Dashboard « scoring crédit »

Projet 7 - Formation Data Scientist

Manuel Martin - 04/2023

Dashboard « scoring crédit »

Agenda

- Problématique / Jeu de données
- Modélisation :
  - Démarche et choix des metrics
  - Tracking des résultats via MLFlow UI
  - Synthèse des résultats
- Pipeline de déploiement
- Analyse de Data Drift
- Dashboard déployé sur le cloud

Problématique / Jeu de données

Problématique / Jeu de données

Problématique

- Besoin d’un outil de « scoring crédit »
  - Probabilité pour qu’un client rembourse son crédit
  - Classification binaire en crédit « accordé » ou « refusé »

- Développer un dashboard interactif

Problématique / Jeu de données

Mission

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

2. Construire un dashboard interactif à destination des gestionnaires de la relation client permettant d'interpréter les prédictions faites par le modèle, et d’améliorer la connaissance client des chargés de relation client.

3. Mettre en production le modèle de scoring de prédiction à l’aide d’une API, ainsi que le dashboard interactif qui appelle l’API pour les prédictions.

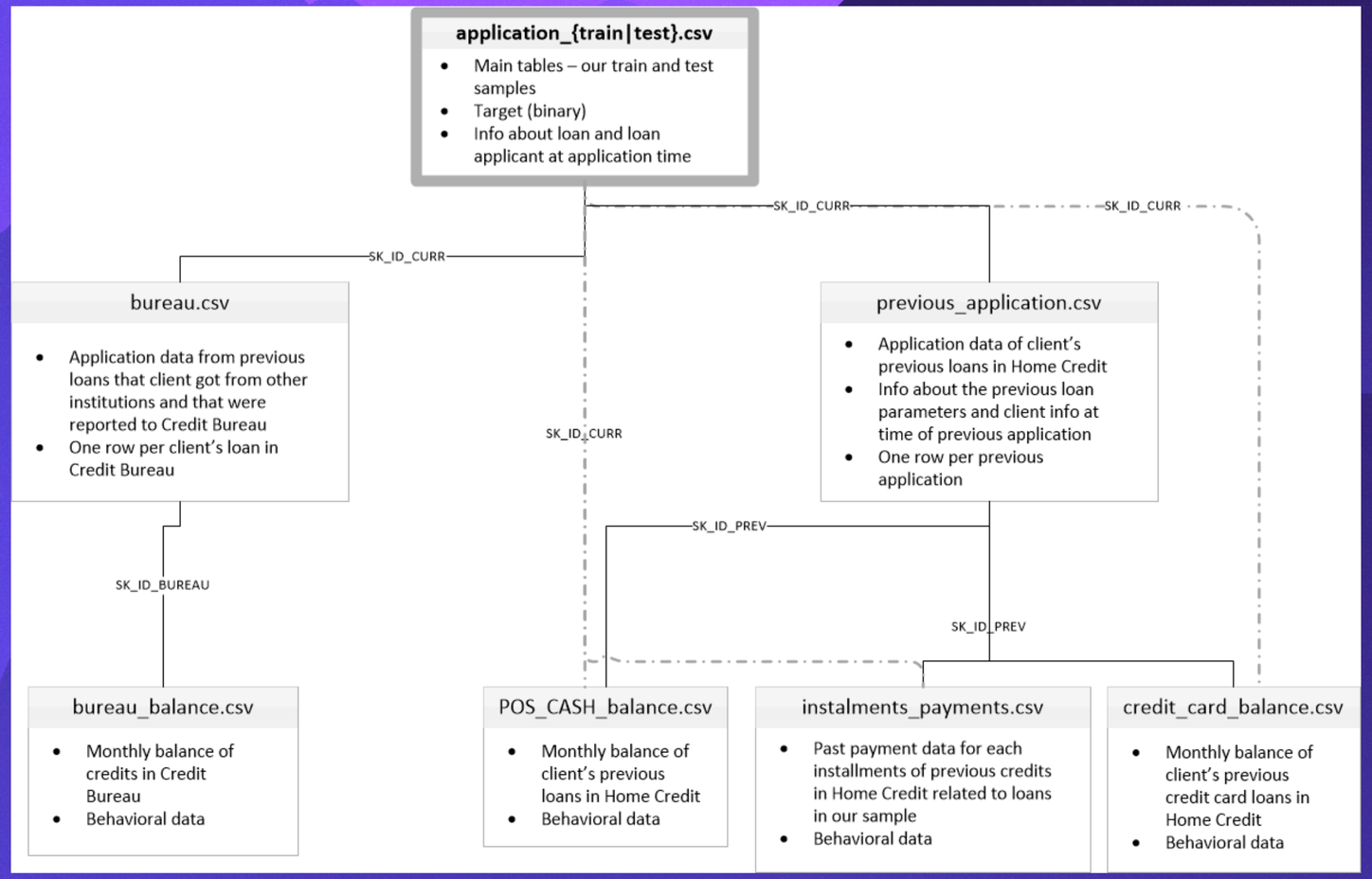
Problématique / Jeu de données

Jeu de données

- Dataset issus du kaggle « Home Credit Default Risk »
- 10 fichiers :
  - 8 dataframes
  - 1 fichier descriptif
  - 1 fichier exemple d’envoi pour la compétition kaggle

Problématique / Jeu de données

Jeu de données



Modélisation

Modélisation

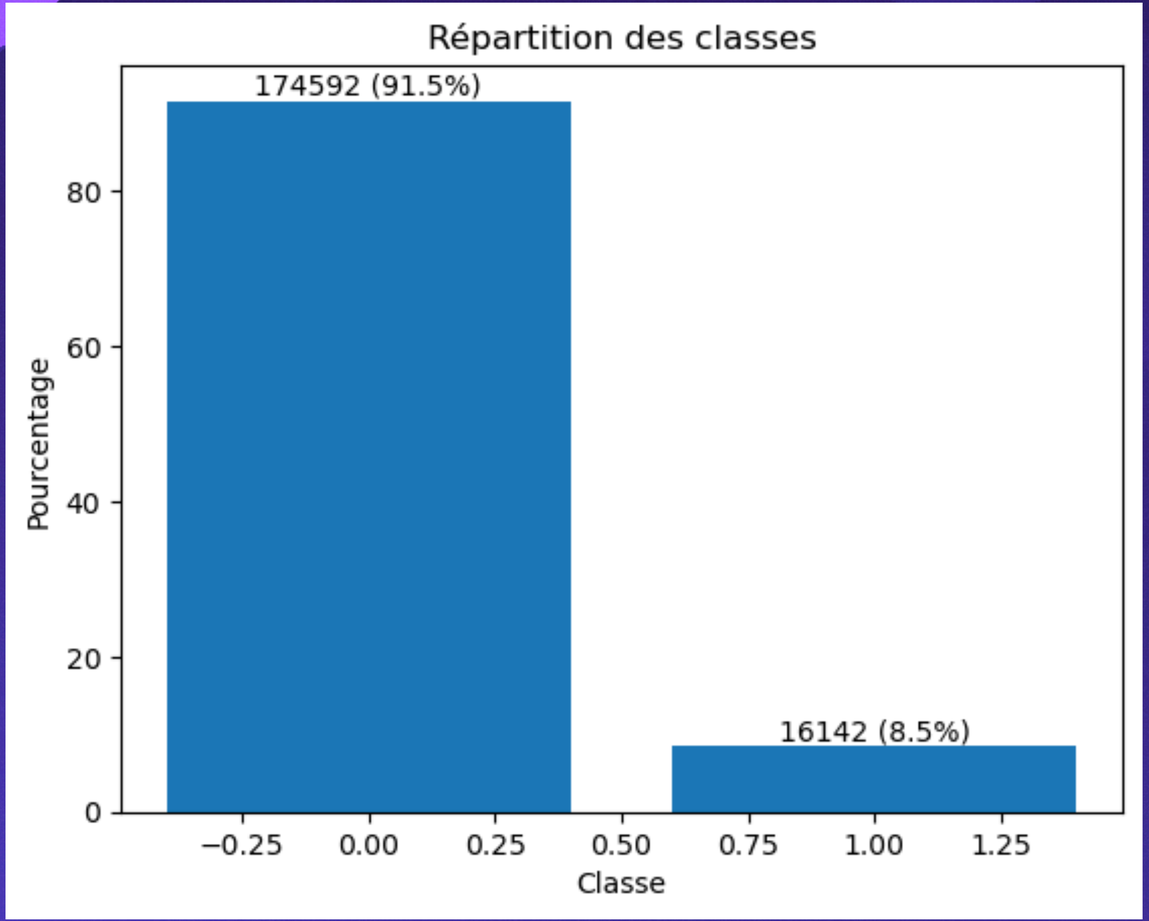
Démarche et choix des metrics

- Pré-traitement des données :
  - Adaptation d’un pré-traitement issu de kaggle
  - Obtention d’un dataframe utilisable pour la modélisation de :
    - 307 507 individus et 464 features
- Sélection des features par corrélation avec la cible :
  - 21 features ayant un coefficient de corrélation > 0,05 en valeur absolue
- Split en bases d’entrainement et de test (20% test)
- Validation croisée en 5 plis (1 pli pour la validation croisée)

Modélisation

Démarche et choix des metrics

- Déséquilibre des classes :
  - Il y a d'avantage de classe 0 (crédit accordé) que de classe 1 (crédit refusé)
  - Rééquilibrage via SMOTE (Synthetic Minority Over-sampling Technique)
  - Pipeline d'Imbalanced-learn pour appliquer SMOTE uniquement sur l'entrainement du modèle



Modélisation

Démarche et choix des metrics

- Validation croisée (sklearn.model_selection.GridSearchCV ou sklearn.model_selection.RandomizedSearchCV) :
  - Pipeline (imblearn.pipeline.Pipeline) :
    - Mise à l'échelle (sklearn.preprocessing.StandardScaler)
    - Traitement du déséquilibre des classes (imblearn.over_sampling.SMOTE)
    - Algorithme de classification binaire

Modélisation

Démarche et choix des metrics

- Algorithmes de classification binaire utilisés :
  - Dummy Classifier (sklearn.dummy.DummyClassifier)
  - Régression Logistique (sklearn.linear_model.LogisticRegression)
  - Forêt Aléatoire (sklearn.ensemble.RandomForestClassifier)
  - LightGBM (lightgbm.LGBMClassifier)

Modélisation

Démarche et choix des metrics

- Problématique métier :
  - Les clients qui ont des problèmes de paiement sont plus importants que les clients qui remboursent.
- Scores métier :
  - F-beta avec beta = 10 : Rappel sera 10 x plus important que la Précision
  - Seuil = 0,45 pour prendre en compte d’abord les FN, ensuite les FP

Modélisation

Démarche et choix des metrics

- Scoring pour la validation croisée :
  - F1 score, F-beta score (beta=10) et Roc-auc score

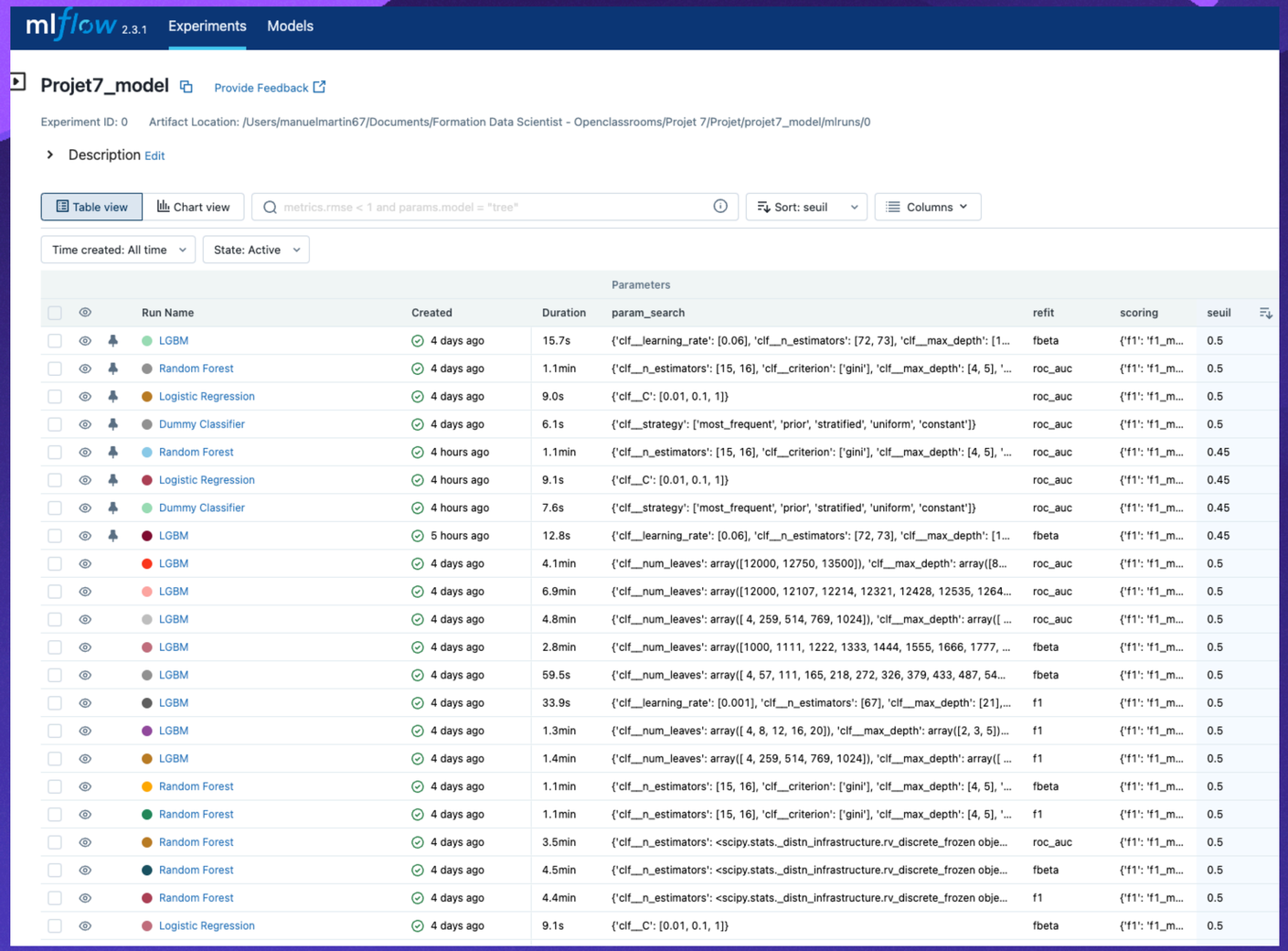
- Seuil : 0.4, 0.45 et 0.5

- Métriques pour base d’entrainement et de test
  - F1 score, F-beta score (beta=10) et Roc-auc score
  - Précision et Rappel

Modélisation

Tracking des résultats via MLFlow UI

- Utilisation de MLFlow pour le tracking :
  - Nom de run = algorithme de classification utilisé
  - Paramètres :
    - Hyperparamètres pour la validation croisée
    - Scoring d'optimisations
    - Score pour réaliser un refit du modèle
    - Seuil pour réaliser les prédictions sur les bases d'entrainement et de test

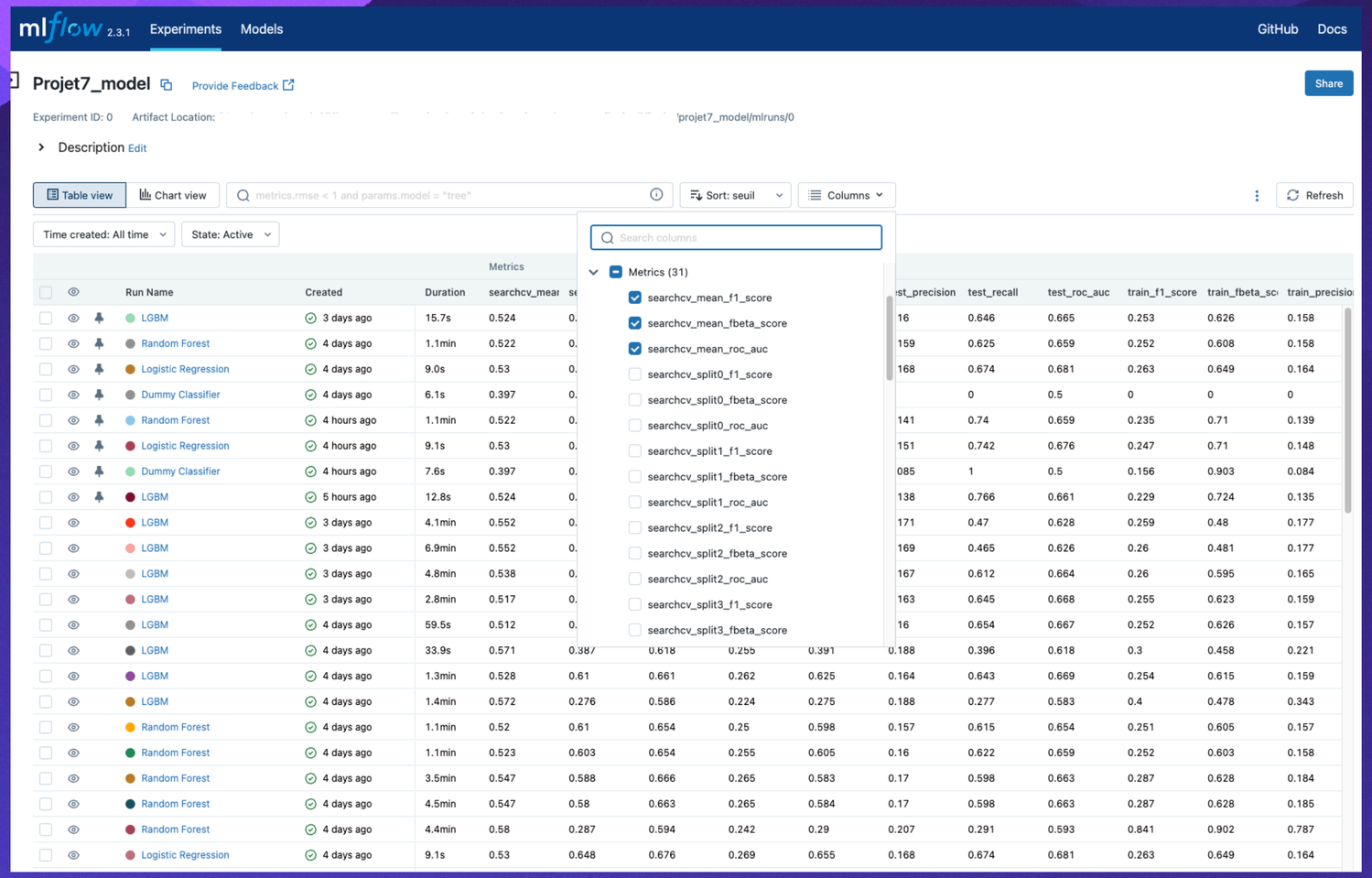


				Parameters			
Run Name	Created	Duration	param_search	refit	scoring	seuil	
LGBM	4 days ago	15.7s	{'clf__learning_rate': [0.06], 'clf__n_estimators': [72, 73], 'clf__max_depth': [1...	fbeta	{'f1': 'f1_m...	0.5	
Random Forest	4 days ago	1.1min	{'clf__n_estimators': [15, 16], 'clf__criterion': ['gini'], 'clf__max_depth': [4, 5], '...	roc_auc	{'f1': 'f1_m...	0.5	
Logistic Regression	4 days ago	9.0s	{'clf__C': [0.01, 0.1, 1]}	roc_auc	{'f1': 'f1_m...	0.5	
Dummy Classifier	4 days ago	6.1s	{'clf__strategy': ['most_frequent', 'prior', 'stratified', 'uniform', 'constant']}	roc_auc	{'f1': 'f1_m...	0.5	
Random Forest	4 hours ago	1.1min	{'clf__n_estimators': [15, 16], 'clf__criterion': ['gini'], 'clf__max_depth': [4, 5], '...	roc_auc	{'f1': 'f1_m...	0.45	
Logistic Regression	4 hours ago	9.1s	{'clf__C': [0.01, 0.1, 1]}	roc_auc	{'f1': 'f1_m...	0.45	
Dummy Classifier	4 hours ago	7.6s	{'clf__strategy': ['most_frequent', 'prior', 'stratified', 'uniform', 'constant']}	roc_auc	{'f1': 'f1_m...	0.45	
LGBM	5 hours ago	12.8s	{'clf__learning_rate': [0.06], 'clf__n_estimators': [72, 73], 'clf__max_depth': [1...	fbeta	{'f1': 'f1_m...	0.45	
LGBM	4 days ago	4.1min	{'clf__num_leaves': array([12000, 12750, 13500]), 'clf__max_depth': array([8...	roc_auc	{'f1': 'f1_m...	0.5	
LGBM	4 days ago	6.9min	{'clf__num_leaves': array([12000, 12107, 12214, 12321, 12428, 12535, 1264...	roc_auc	{'f1': 'f1_m...	0.5	
LGBM	4 days ago	4.8min	{'clf__num_leaves': array([4, 259, 514, 769, 1024]), 'clf__max_depth': array([...	roc_auc	{'f1': 'f1_m...	0.5	
LGBM	4 days ago	2.8min	{'clf__num_leaves': array([1000, 1111, 1222, 1333, 1444, 1555, 1666, 1777, ...	fbeta	{'f1': 'f1_m...	0.5	
LGBM	4 days ago	59.5s	{'clf__num_leaves': array([4, 57, 111, 165, 218, 272, 326, 379, 433, 487, 54...	fbeta	{'f1': 'f1_m...	0.5	
LGBM	4 days ago	33.9s	{'clf__learning_rate': [0.001], 'clf__n_estimators': [67], 'clf__max_depth': [21],...	f1	{'f1': 'f1_m...	0.5	
LGBM	4 days ago	1.3min	{'clf__num_leaves': array([4, 8, 12, 16, 20]), 'clf__max_depth': array([2, 3, 5])...	f1	{'f1': 'f1_m...	0.5	
LGBM	4 days ago	1.4min	{'clf__num_leaves': array([4, 259, 514, 769, 1024]), 'clf__max_depth': array([...	f1	{'f1': 'f1_m...	0.5	
Random Forest	4 days ago	1.1min	{'clf__n_estimators': [15, 16], 'clf__criterion': ['gini'], 'clf__max_depth': [4, 5], '...	fbeta	{'f1': 'f1_m...	0.5	
Random Forest	4 days ago	1.1min	{'clf__n_estimators': [15, 16], 'clf__criterion': ['gini'], 'clf__max_depth': [4, 5], '...	f1	{'f1': 'f1_m...	0.5	
Random Forest	4 days ago	3.5min	{'clf__n_estimators': <scipy.stats._distn_infrastructure.rv_discrete_frozen obje...	roc_auc	{'f1': 'f1_m...	0.5	
Random Forest	4 days ago	4.5min	{'clf__n_estimators': <scipy.stats._distn_infrastructure.rv_discrete_frozen obje...	fbeta	{'f1': 'f1_m...	0.5	
Random Forest	4 days ago	4.4min	{'clf__n_estimators': <scipy.stats._distn_infrastructure.rv_discrete_frozen obje...	f1	{'f1': 'f1_m...	0.5	
Logistic Regression	4 days ago	9.1s	{'clf__C': [0.01, 0.1, 1]}	fbeta	{'f1': 'f1_m...	0.5	

Modélisation

Tracking des résultats via MLFlow UI

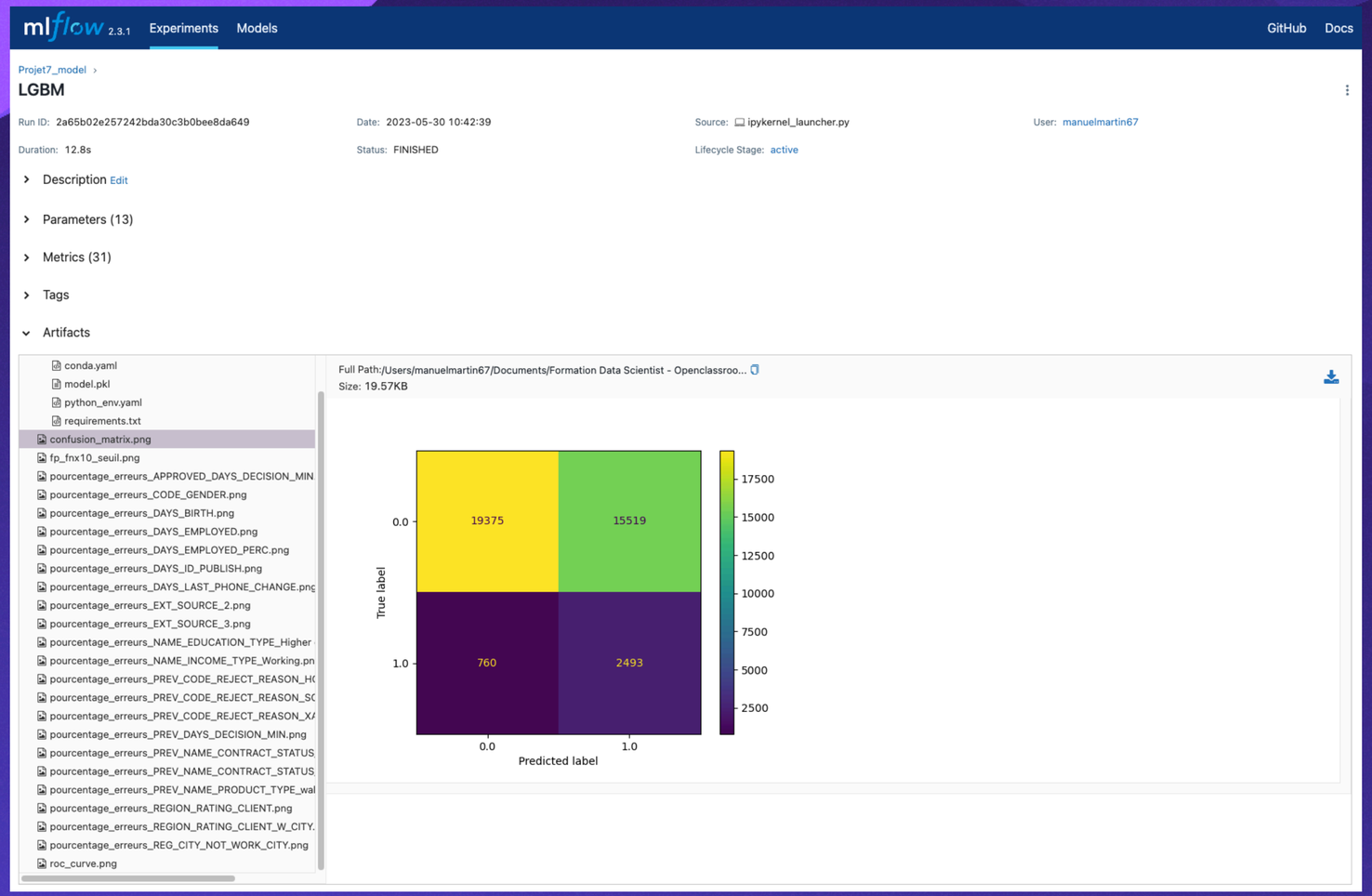
- Utilisation de MLFlow pour le tracking :
  - Métriques :
    - Scores de chaque pli et scores moyens pour les 3 scoring d'évaluations
    - Scores des prédictions sur les bases d'entrainement et de test des 5 métriques en fonction du seuil



Modélisation

Tracking des résultats via MLFlow UI

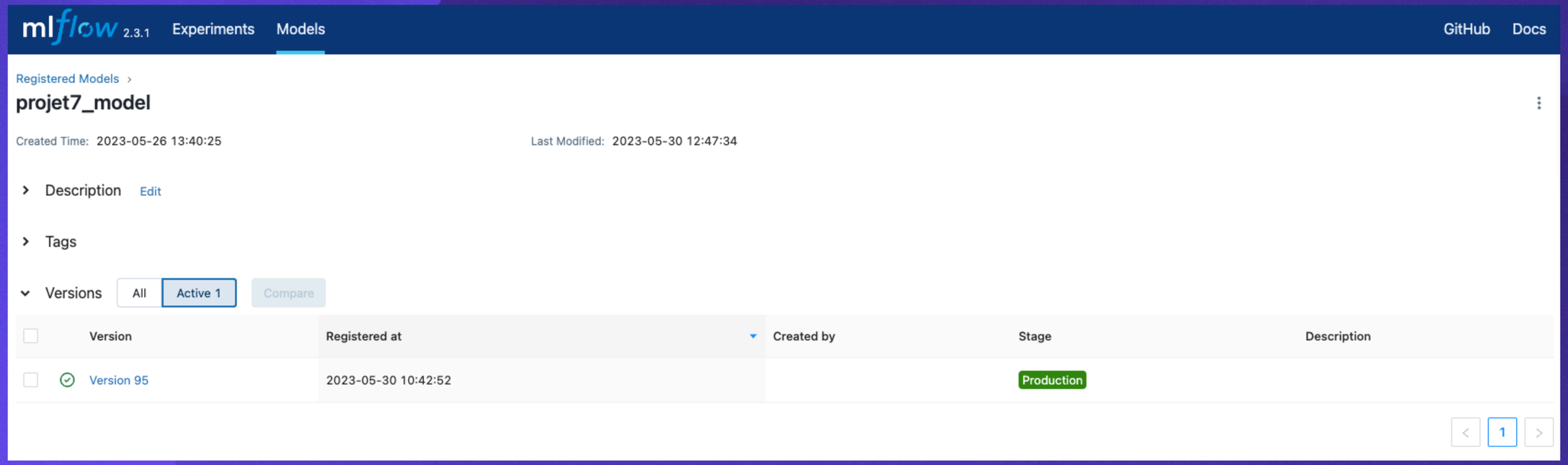
- Utilisation de MLFlow pour le tracking :
  - Meilleur modèle de la validation croisée
  - Graphiques :
    - Courbe ROC
    - FP et FN(x10) en fonction du seuil
    - Pourcentage d’erreur en fonction des valeurs des features
    - Matrice de confusion



Modélisation

Tracking des résultats via MLFlow UI

- Utilisation de MLFlow Models Registry
  - Sélection et tracking des modèles, et export des modèles pour serving



Modélisation

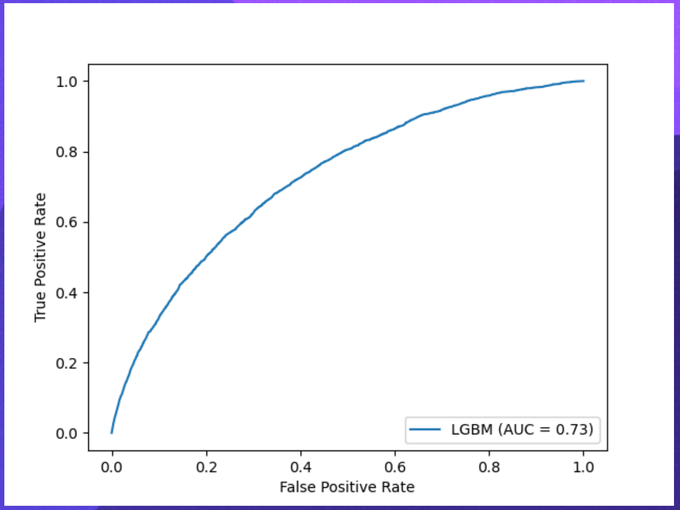
Synthèse des résultats

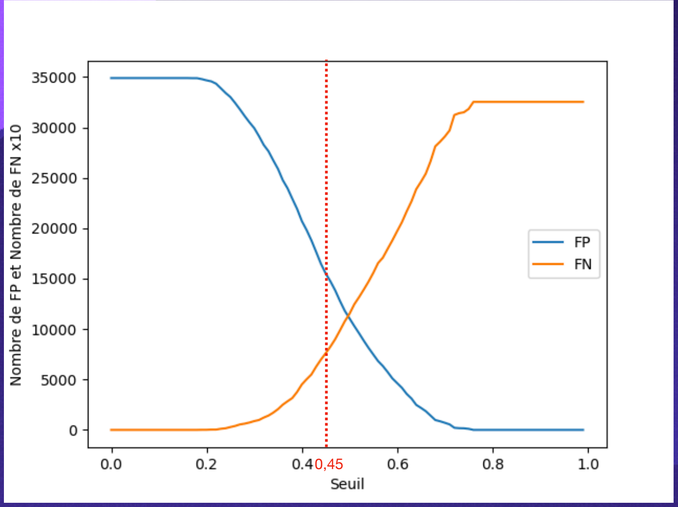
			Metrics									Parameters	
Run Name	Created	Duration	searchcv_mear	searchcv_mean_fbeta_score	searchcv_mear	test_f1_score	test_fbeta_score	test_roc_auc	train_f1_score	train_fbeta_sc	train_roc_auc	refit	seuil
Dummy Classifier	2 days ago	7.3s	0.397	0.483	0.504	0.157	0.904	0.5	0.156	0.903	0.5	fbeta	0.45
LGBM	2 days ago	12.8s	0.524	0.623	0.662	0.234	0.733	0.661	0.229	0.724	0.655	fbeta	0.45
Logistic Regression	2 days ago	9.4s	0.53	0.648	0.676	0.251	0.715	0.676	0.247	0.71	0.673	fbeta	0.45
Random Forest	2 days ago	1.1min	0.52	0.61	0.654	0.235	0.704	0.657	0.234	0.705	0.658	fbeta	0.45
Logistic Regression	5 days ago	9.1s	0.53	0.648	0.676	0.269	0.655	0.681	0.263	0.649	0.677	fbeta	0.5
LGBM	5 days ago	15.7s	0.524	0.623	0.662	0.256	0.627	0.665	0.253	0.626	0.664	fbeta	0.5
Random Forest	5 days ago	1.1min	0.52	0.61	0.654	0.25	0.598	0.654	0.251	0.605	0.658	fbeta	0.5
Dummy Classifier	5 days ago	6.3s	0.397	0.483	0.504	0	0	0.5	0	0	0.5	fbeta	0.5

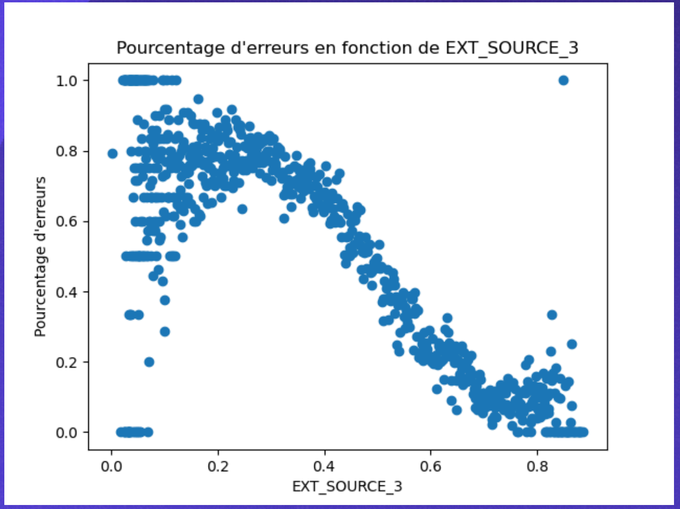
- Meilleur modèle avec un seuil à 0,5 : Regression Logistique
- Meilleur modèle avec un seuil à 0,45 : LGBM
    - Modèle retenu : LGBM et seuil à 0,45

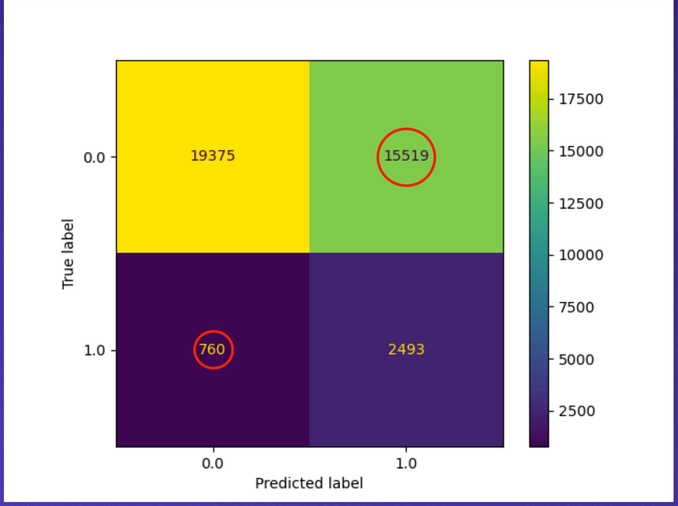
Modélisation

Synthèse des résultats







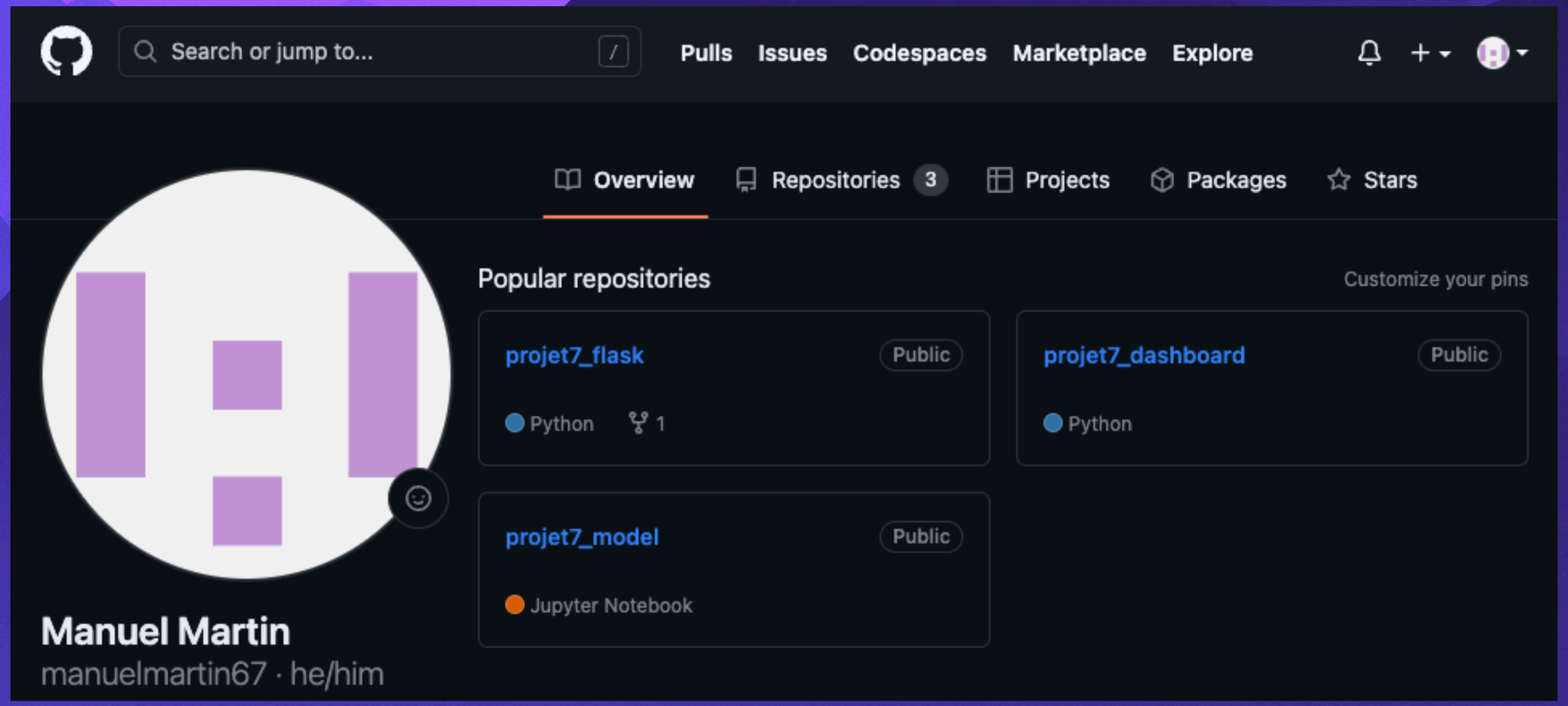


Pipeline de déploiement

Pipeline de déploiement

Git, Github, tests unitaires

- 3 repositories :
  - projet7_model
    https://github.com/manuelmartin67/projet7_model
  - projet7_flask
    https://github.com/manuelmartin67/projet7_flask
  - projet7_dashboard
    https://github.com/manuelmartin67/projet7_dashboard

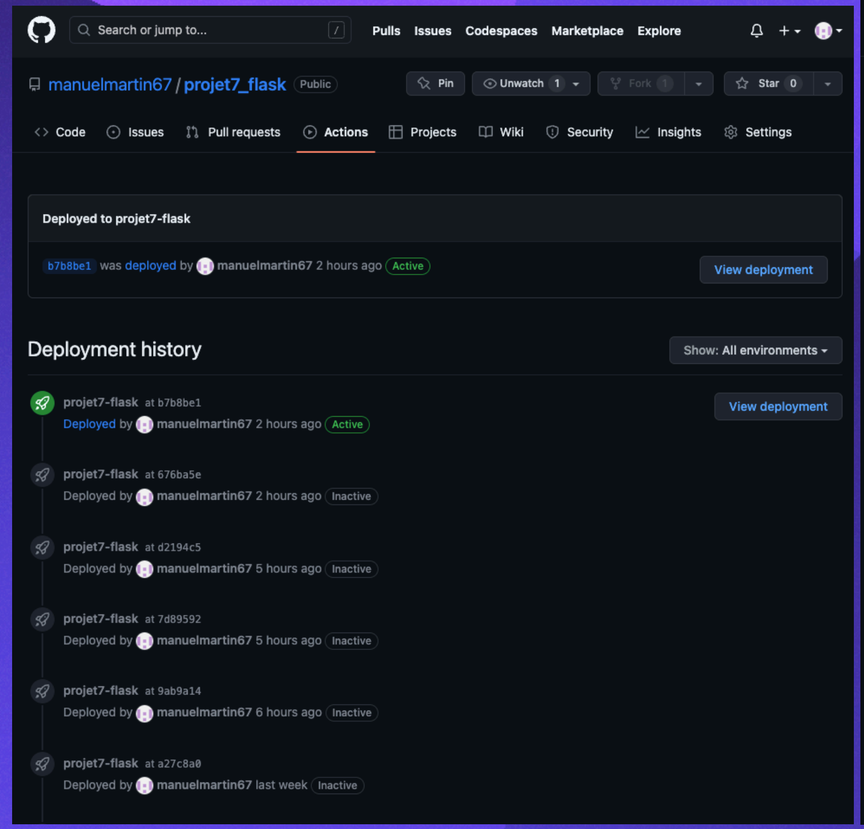


Capture d'écran des 3 repositories (dossiers) de ce projet

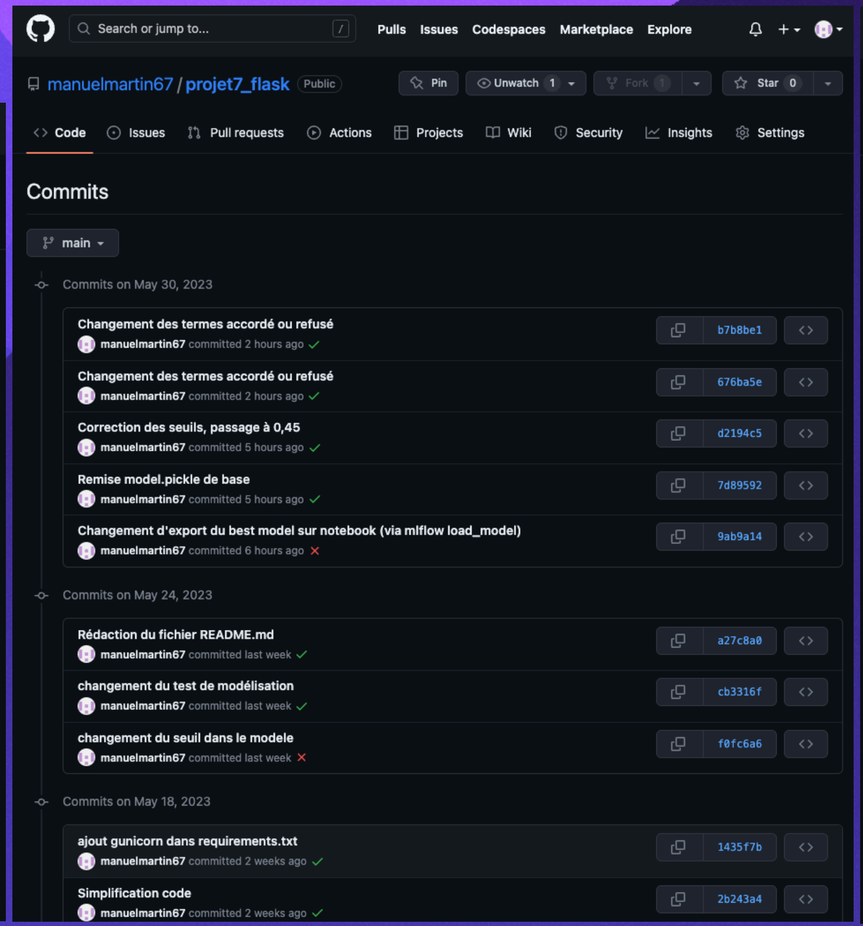
Pipeline de déploiement

Git, Github, tests unitaires

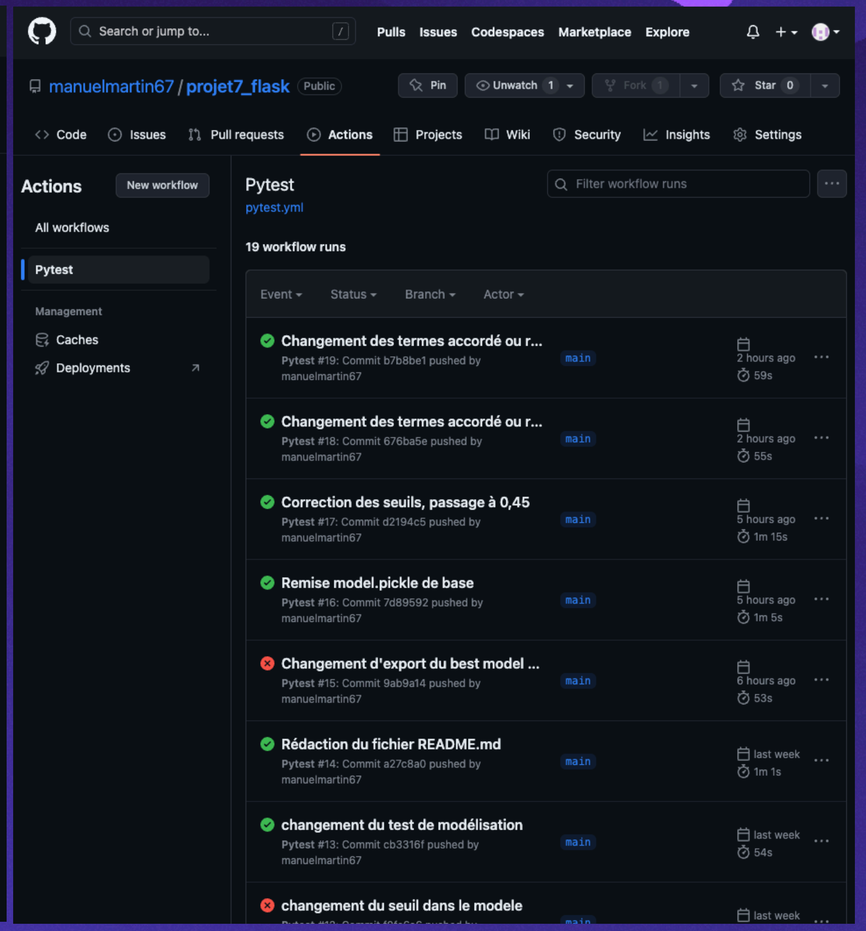
- projet7_flask :



Capture d'écran des déploiements pour Heroku



Capture d'écran des commits de l'API



Capture d'écran des workflows de tests via Pytest

Analyse de Data Drift

Analyse de Data Drift

- Data Drift :
  - Divergence au fil du temps de la base de données
  - Trop de data drift peut amener à une diminution de la performance et la fiabilité du modèle
- Librairie python : Evidently (https://docs.evidentlyai.com/reference/api-reference)
  - evidently.metrics.data_drift

Analyse de Data Drift

- Analyse entre Dataset Train et Test pré-traités
- 2 features présentent du drift
  - Drift score > 0,1
  - Presque 10% des features utilisées par le modèle

Dataset Drift

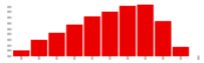
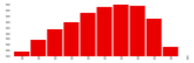
Dataset Drift is NOT detected. Dataset drift detection threshold is 0.5

21	2	0.0952
Columns	Drifted Columns	Share of Drifted Columns

Data Drift Summary

Drift is detected for 9.524% of columns (2 out of 21).

Search

Column	Type	Reference Distribution	Current Distribution	Data Drift	Stat Test	Drift Score
DAYS_LAST_PHONE_CHANGE	num			Detected	Wasserstein distance (normed)	0.110647
PREV_NAME_CONTRACT_STATUS_Approved_MEAN	num			Detected	Wasserstein distance (normed)	0.109714
EXT_SOURCE_3	num			Not Detected	Wasserstein distance (normed)	0.056587
PREV_CODE_REJECT_REASON_HC_MEAN	num			Not Detected	Wasserstein distance (normed)	0.054691

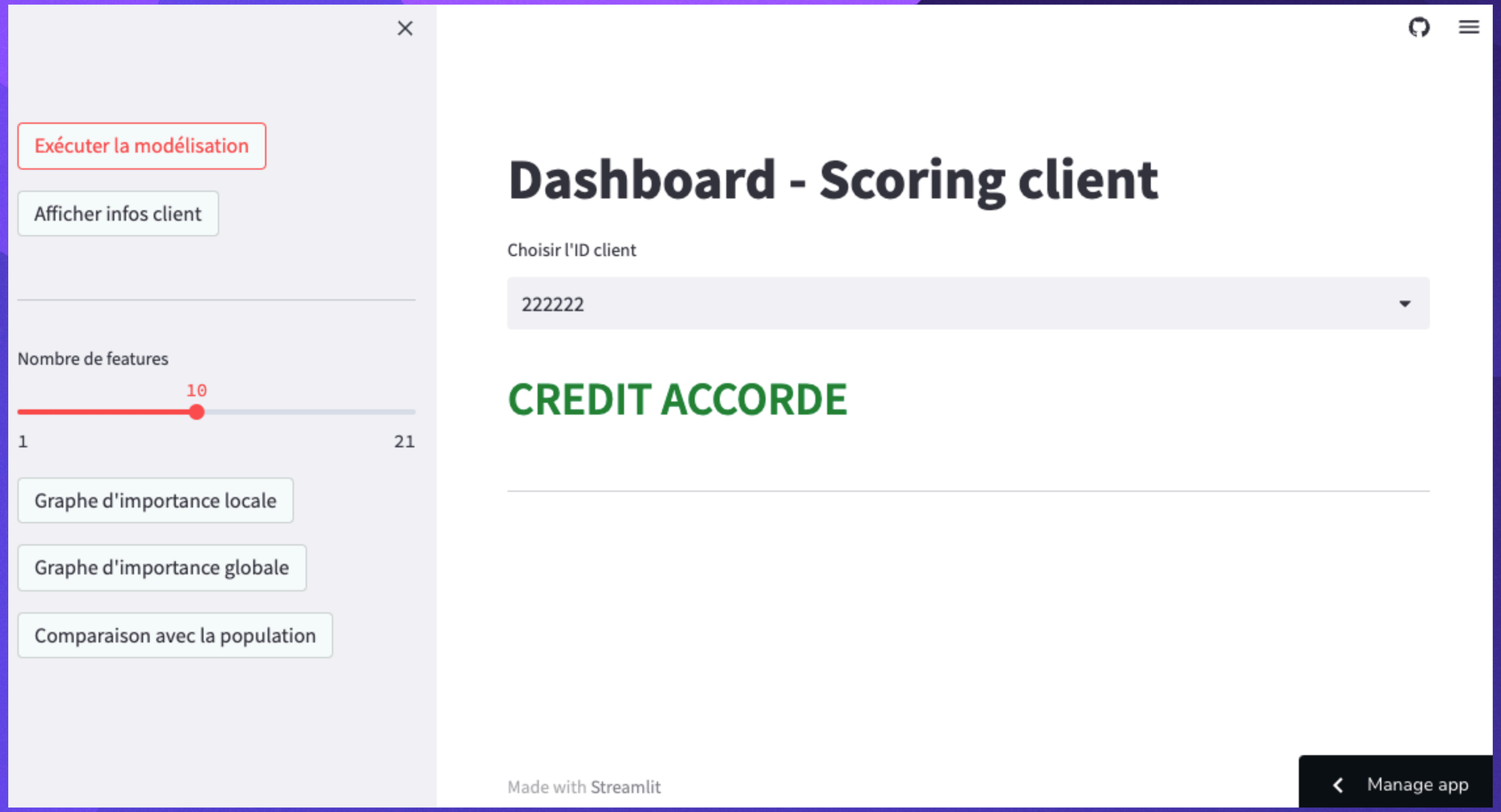
Dashboard déployé sur le cloud

Dashboard déployé sur le cloud

- Dashboard réalisé avec la librairie et déployé sur Streamlit
  - https://manuelmartin67-projet7-dashboard-app-843ltm.streamlit.app/
  - Prise en compte de critères d’accessibilité du WCAG 2

- Requêtes vers API réalisée avec la librairie Flask, et déployée sur Heroku
  - https://projet7-flask.herokuapp.com/

Dashboard déployé sur le cloud



Merci