

# Mise en place d'un entraînement contin avec Jenkins

## Projet MLOps

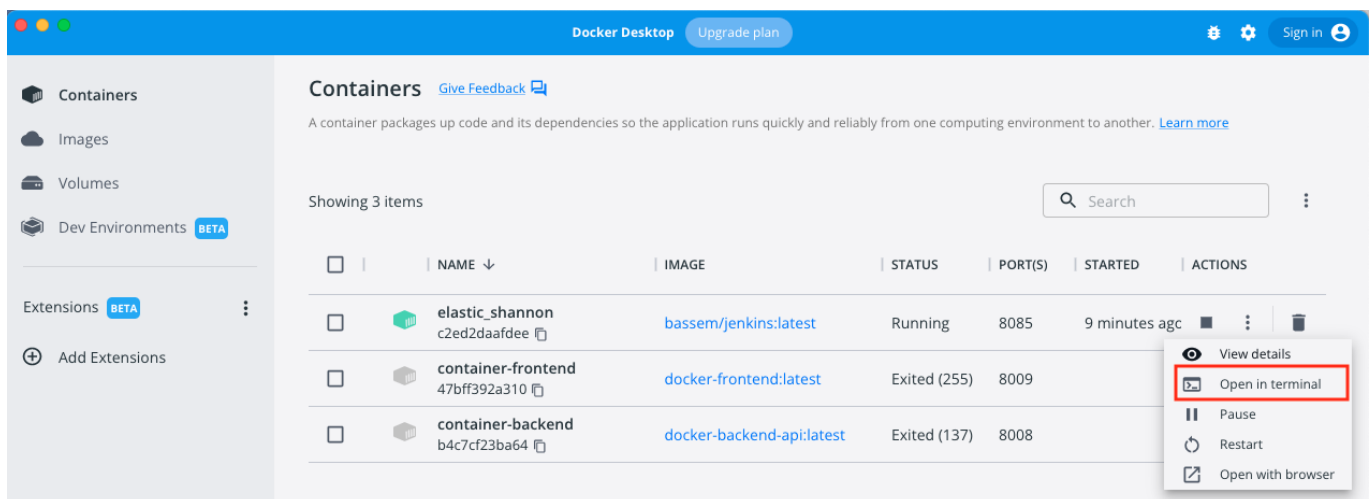
Les modèles ML se périment avec le temps et leurs performances se dégradent. Cela est dû au fait que les données changent en temps réel en raison de l'évolution des tendances. C'est ce qu'on appelle **Data Drift**. Si on laisse passer suffisamment de temps, il est également possible que la relation entre les features et les targets elle-même change. C'est ce qu'on appelle **Concept Drift**. Pour rester pertinent par rapport aux données, on devra maintenir les modèles ML qu'on utilise à jour par rapport aux données.

L'entraînement continu (CT) est le recyclage automatique des modèles ML afin de les redéployer.

Jenkins est le principal serveur d'automatisation open source qui propose des centaines de plugins qui permettent de construire, de déployer et d'automatiser n'importe quel projet.

Les étapes de mise en place de jenkins dans notre projet MLOps sont les suivantes :

1. Ouvrir l'image Jenkins dans un terminal.



```
$ pip install dagshub
```

```
$ pip install pandas
```

```
$ pip install mlflow
```

```
$ pip install imblearn
```

```
$ pip install python-dotenv
```

```
$ pip install xgboost
```

2. Accéder à votre serveur Jenkins. Pour créer un nouveau job, choisir **NewItem**, puis nommer le par exemple **CT\_job** et sélectionner **Pipeline**.

## Pipeline

### Definition

Pipeline script

#### Script ?

```
1 pipeline {
2   agent any
3
4   stages {
5     stage('Checkout') {
6       steps {
7         checkout([class: 'GitSCM', branches: [[name: '**']], extensions: [], userRemoteConfigs: [[credentialsId: 'Dagshub credential', url: 'https://dagshub.com/bassem.benham
8       }
9     }
10  }
11  stage('Python script') {
12    steps {
13      sh 'python3 ./backend/src/modeling.py'
14    }
15  }
16 }
```

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

3. Dans la section **Steps** de **Pipeline Syntax**, sélectionner l'option **Checkout: Check out from version control**, recopier l'URL de votre repository Dagshub et choisir le **Credentials**

localhost:8085/job/job1/pipeline-syntax/

Dashboard > job1 > Pipeline Syntax

**Steps**

Sample Step

checkout: Check out from version control

checkout ?

SCM

Git

Repositories ?

Repository URL ?

https://dagshub.com/bassem.benham/mlops\_project.git

Credentials ?

bassem.benhamed/\*\*\*\*\*

+ Add

Advanced...

8. Cliquer sur **Generate Pipeline Script**, puis le recopier dans le **Jenkinsfile**.

9. **Save**, puis **Build Now**.

# Pipeline CTJob

## Stage View

