

# MLOps en pratique : Construire et suivre vos modèles IA



**Arthur Cartel Foahom  
Souabou**  
Data Scientist & Chercheur en IA | Formateur IT

# Formateur

---



# Formateur

---

Arthur Cartel FOAHOM GOUABOU



ALPHORM.COM  
  
**MLOps**

# Formateur

---

**Arthur Cartel FOAHOM GOUABOU**

- Chercheur en IA & Data Scientist senior



# Formateur

---

## Arthur Cartel FOAHOM GOUABOU

- Chercheur en IA & Data Scientist senior
- 7 années d'expérience en IA et Deep Learning



# Formateur

---

## Arthur Cartel FOAHOM GOUABOU

- Chercheur en IA & Data Scientist senior
- 7 années d'expérience en IA et Deep Learning
- Projets menés dans la santé, la logistique et l'industrie



ALPHORM.COM

∞  
**MLOps**

# Formateur

---

## Arthur Cartel FOAHOM GOUABOU

- Chercheur en IA & Data Scientist senior
- 7 années d'expérience en IA et Deep Learning
- Projets menés dans la santé, la logistique et l'industrie
- Formateur, mentor et auteur de publications scientifiques internationales



ALPHORM.COM

∞  
**MLOps**

# Rappel cursus MLOps

# Rappel cursus MLOps



# Rappel cursus MLOps



# Rappel cursus MLOps



# Plan de la formation

1. Découvrir le MLOps et préparer l'environnement

# Plan de la formation

1. Découvrir le MLOps et préparer l'environnement
2. Organiser un projet ML reproductible

# Plan de la formation

1. Découvrir le MLOps et préparer l'environnement
2. Organiser un projet ML reproductible
3. Suivre les expériences (MLflow) et gérer les données (DVC)

# Objectifs pédagogiques

# Objectifs pédagogiques

- Comprendre et appliquer les bonnes pratiques pour structurer un projet ML

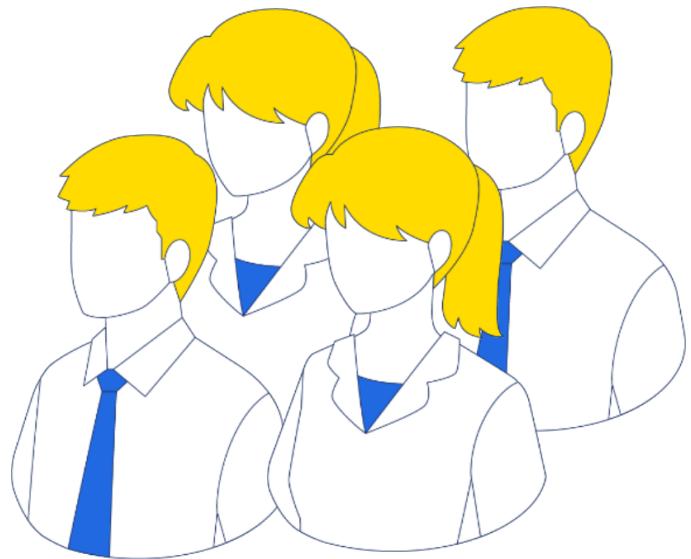
# Objectifs pédagogiques

- Comprendre et appliquer les bonnes pratiques pour structurer un projet ML
- Automatiser les étapes d'ingestion, d'entraînement et d'évaluation des modèles

# Objectifs pédagogiques

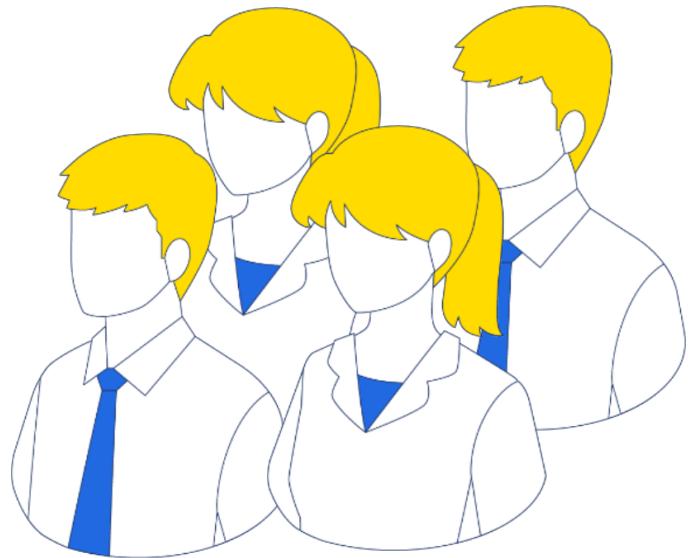
- Comprendre et appliquer les bonnes pratiques pour structurer un projet ML
- Automatiser les étapes d'ingestion, d'entraînement et d'évaluation des modèles
- Suivre les expérimentations et gérer les versions des données et modèles

# Public concerné



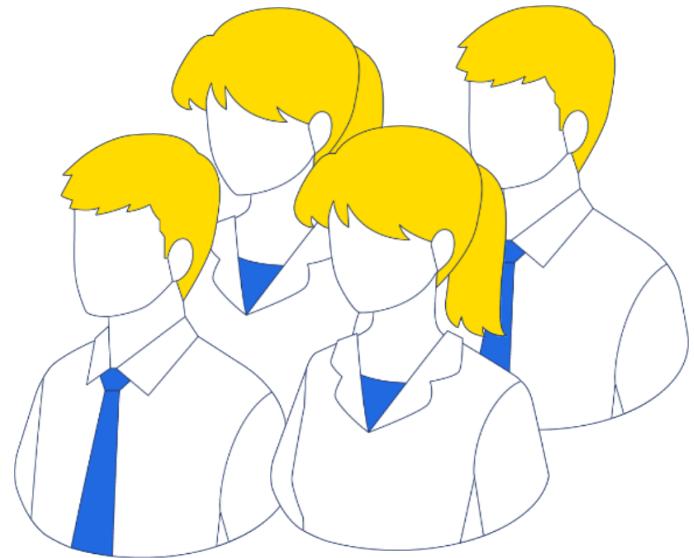
# Public concerné

- Data Scientist ou ML Engineer



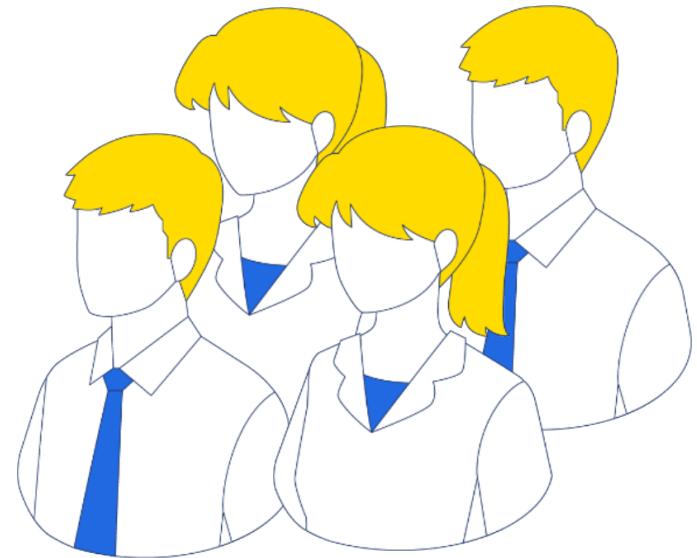
# Public concerné

- Data Scientist ou ML Engineer
- Ingénieur DevOps



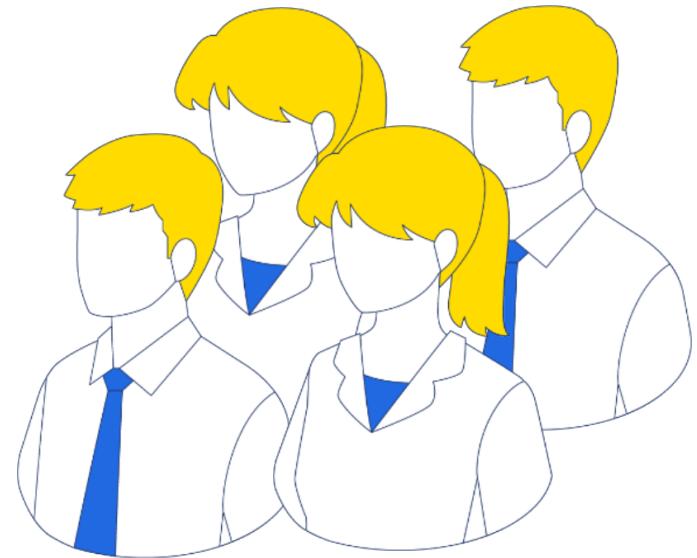
# Public concerné

- Data Scientist ou ML Engineer
- Ingénieur DevOps
- Étudiants ou passionnés de ML



# Public concerné

- Data Scientist ou ML Engineer
- Ingénieur DevOps
- Étudiants ou passionnés de ML
- Professionnels appliquant l'IA en entreprise



# Connaissances requises

# Connaissances requises



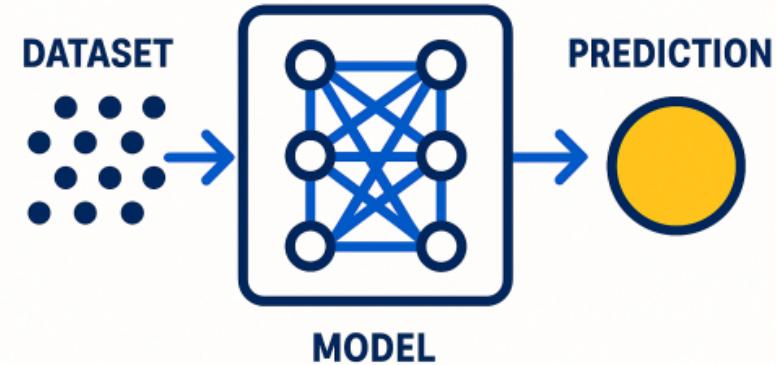
Bases en  
**Python**

# Connaissances requises

---



Bases en  
Python

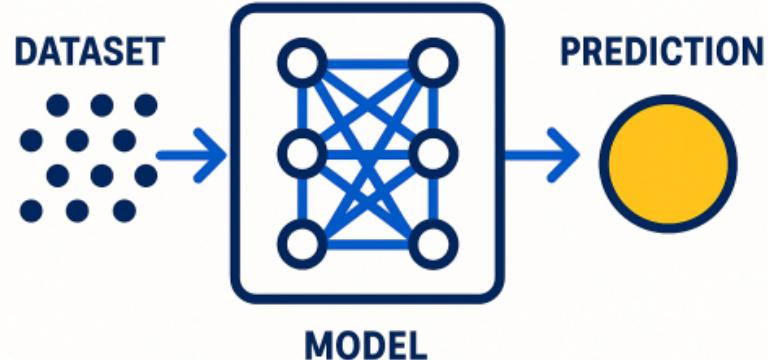


Notions en **Machine  
Learning**

# Connaissances requises



Bases en  
Python



Notions en **Machine  
Learning**



Confortable avec un **environnement de développement**  
**(VS code, Git, Terminal)**



# Présentation du projet



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Présentation du projet fil rouge

# Plan

- § Présentation du projet fil rouge
- § Outils utilisés

# Plan

- § Présentation du projet fil rouge
- § Outils utilisés
- § Pourquoi ce projet est pertinent ?

# Plan

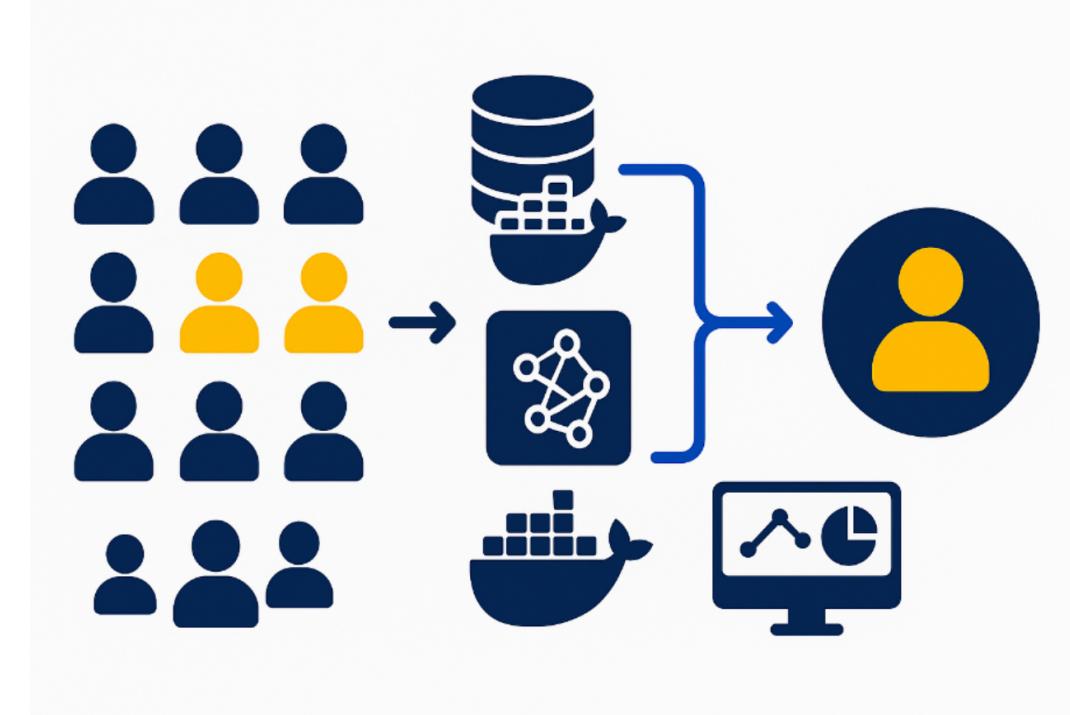
- § Présentation du projet fil rouge
- § Outils utilisés
- § Pourquoi ce projet est pertinent ?
- § Ce que vous allez apprendre

# Plan

- § Présentation du projet fil rouge
- § Outils utilisés
- § Pourquoi ce projet est pertinent ?
- § Ce que vous allez apprendre
- § Démonstration

# Présentation du projet

# Présentation du projet



Prédiction du churn client

# Présentation du projet

- Cas métier concret : prédiction du churn client



Prédiction du churn client

# Présentation du projet

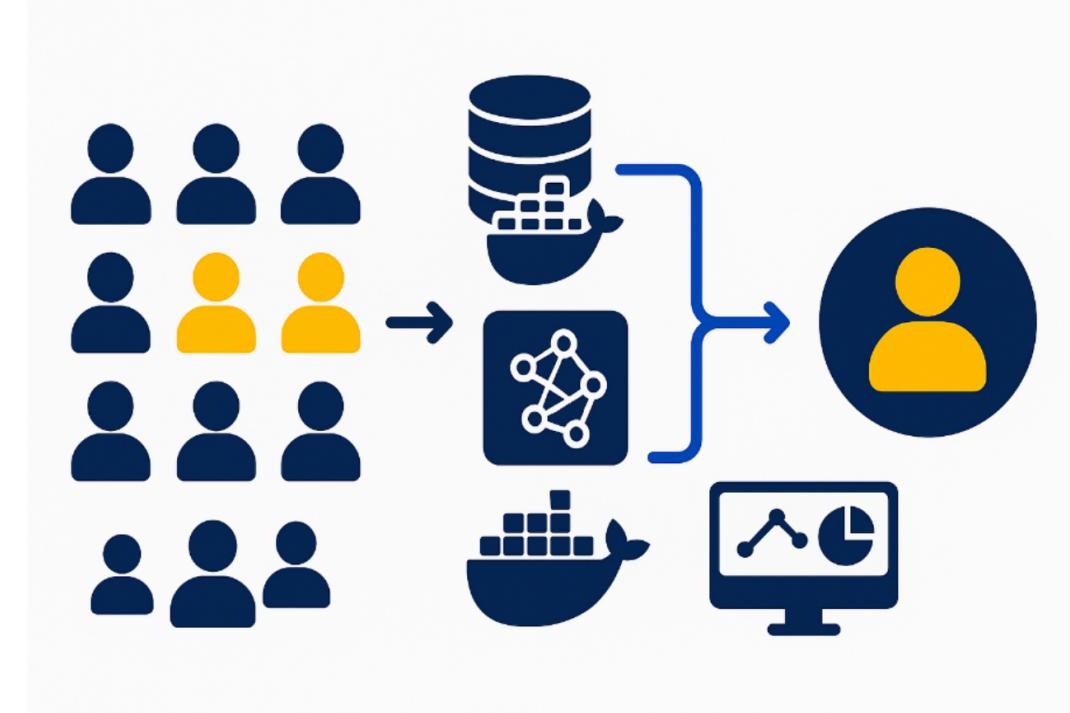
- Cas métier concret : prédiction du churn client
- De la donnée brute au pipeline reproductible



Prédiction du churn client

# Présentation du projet

- Cas métier concret : prédiction du churn client
- De la donnée brute au pipeline reproductible
- Fil conducteur de la formation



Prédiction du churn client

# Présentation des outils

# Présentation des outils

- Python, VSCode, Git/GitHub : environnement de travail



# Présentation des outils

- Python, VSCode, Git/GitHub : environnement de travail
- Linux + SFTP : ingestion et sécurisation des données



# Présentation des outils

- Python, VSCode, Git/GitHub : environnement de travail
- Linux + SFTP : ingestion et sécurisation des données
- MLflow : suivi des expériences



# Présentation des outils

- Python, VSCode, Git/GitHub : environnement de travail
- Linux + SFTP : ingestion et sécurisation des données
- MLflow : suivi des expériences
- DVC : gestion et versioning des données



Pourquoi ce projet est pertinent  
?

# Pourquoi ce projet est pertinent ?

- Répond aux vrais besoins de structuration en ML

# Pourquoi ce projet est pertinent ?

- Répond aux vrais besoins de structuration en ML
- Met en place un environnement stable et reproduitble

# Pourquoi ce projet est pertinent ?

- Répond aux vrais besoins de structuration en ML
- Met en place un environnement stable et reproductible
- Introduit des outils clés pour gérer expériences et données  
(MLflow & DVC)

# Pourquoi ce projet est pertinent ?

- Répond aux vrais besoins de structuration en ML
- Met en place un environnement stable et reproduitible
- Introduit des outils clés pour gérer expériences et données  
(MLflow & DVC)
- Prépare le terrain pour le déploiement (Partie 2)

# Ce que vous allez apprendre

- Comprendre les concepts du MLOps et le cycle de vie ML

# Ce que vous allez apprendre

- Comprendre les concepts du MLOps et le cycle de vie ML
- Installer un environnement complet et structurer un projet reproductible

# Ce que vous allez apprendre

- Comprendre les concepts du MLOps et le cycle de vie ML
- Installer un environnement complet et structurer un projet reproductible
- Suivre, comparer et gérer vos expériences avec Mlflow

# Ce que vous allez apprendre

- Comprendre les concepts du MLOps et le cycle de vie ML
- Installer un environnement complet et structurer un projet reproductible
- Suivre, comparer et gérer vos expériences avec Mlflow
- Gérer et versionner vos données efficacement avec DVC

# Démonstration

---



# En résumé

# En résumé

- Bases solides du MLOps posées

# En résumé

- Bases solides du MLOps posées
- Projet structuré et reproductible

# En résumé

- Bases solides du MLOps posées
- Projet structuré et reproductible
- Outils professionnels déjà maîtrisés

# Identifier les différences entre la Data Science et le MLOps



Arthur Cartel Foahom Gouabou

Data Scientist & Chercheur en IA | Formateur IT

# Plan

- § Définir le MLOps
- § Pourquoi le MLOps ?
- § Enjeux et bénéfices du MLOps

# Définition du MLOps

MLOps = Machine Learning + Operations

# Définition du MLOps

MLOps = Machine Learning + Operations

Ensemble de **pratiques et outils** pour **concevoir, déployer et maintenir** un modèle de machine learning en production de manière **fiable et continue**.

# Définition du MLOps

MLOps = Machine Learning + Operations

Ensemble de **pratiques et outils** pour **concevoir, déployer et maintenir** un modèle de machine learning en production de manière **fiable et continue**.

“... modèle .... en **production** ...”

# Pourquoi le MLOps?



# Pourquoi le MLOps?

## Reproductibilité des modèles

Difficile de reproduire un modèle sur un autre environnement ou une autre machine.



## Versioning et suivi

Absence de traçabilité des versions de modèles, de données et d'expériences.



## Déploiement en production

Passage complexe du prototype local à une application fiable en environnement réel.

## Robustesse

Modèles instables ou dégradant leurs performances sans mécanismes de contrôle et de maintenance.



# Pourquoi le MLOps?

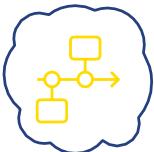
## Reproductibilité des modèles

Difficile de reproduire un modèle sur un autre environnement ou une autre machine.



## Versioning et suivi

Absence de traçabilité des versions de modèles, de données et d'expériences.



## Déploiement en production

Passage complexe du prototype local à une application fiable en environnement réel.

## Robustesse

Modèles instables ou dégradant leurs performances sans mécanismes de contrôle et de maintenance.



# Pourquoi le MLOps?

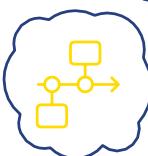
## Reproductibilité des modèles

Difficile de reproduire un modèle sur un autre environnement ou une autre machine.



## Versioning et suivi

Absence de traçabilité des versions de modèles, de données et d'expériences.



## Déploiement en production

Passage complexe du prototype local à une application fiable en environnement réel.

## Robustesse

Modèles instables ou dégradant leurs performances sans mécanismes de contrôle et de maintenance.

# Pourquoi le MLOps?

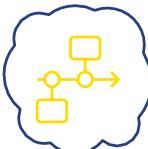
## Reproductibilité des modèles

Difficile de reproduire un modèle sur un autre environnement ou une autre machine.



## Versioning et suivi

Absence de traçabilité des versions de modèles, de données et d'expériences.



## Déploiement en production

Passage complexe du prototype local à une application fiable en environnement réel.

## Robustesse

Modèles instables ou dégradant leurs performances sans mécanismes de contrôle et de maintenance.



# Pourquoi le MLOps?

## Reproductibilité des modèles

Difficile de reproduire un modèle sur un autre environnement ou une autre machine.



## Versioning et suivi

Absence de traçabilité des versions de modèles, de données et d'expériences.



## Déploiement en production

Passage complexe du prototype local à une application fiable en environnement réel.

## Robustesse

Modèles instables ou dégradant leurs performances sans mécanismes de contrôle et de maintenance.



# Pourquoi le MLOps?

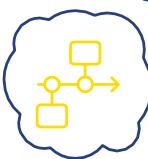
## Reproductibilité des modèles

Difficile de reproduire un modèle sur un autre environnement ou une autre machine.



## Versioning et suivi

Absence de traçabilité des versions de modèles, de données et d'expériences.



## Déploiement en production

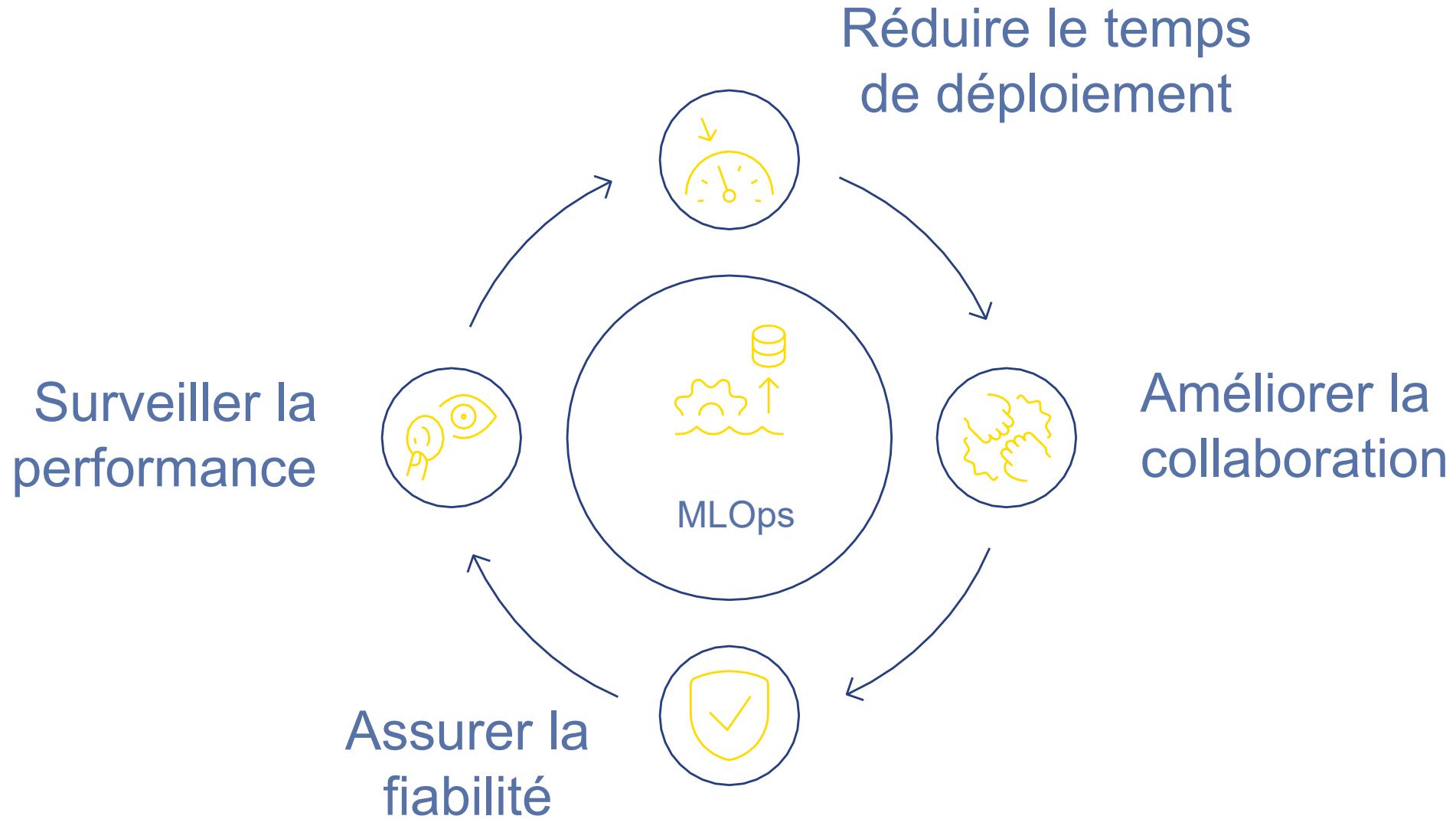
Passage complexe du prototype local à une application fiable en environnement réel.

## Robustesse

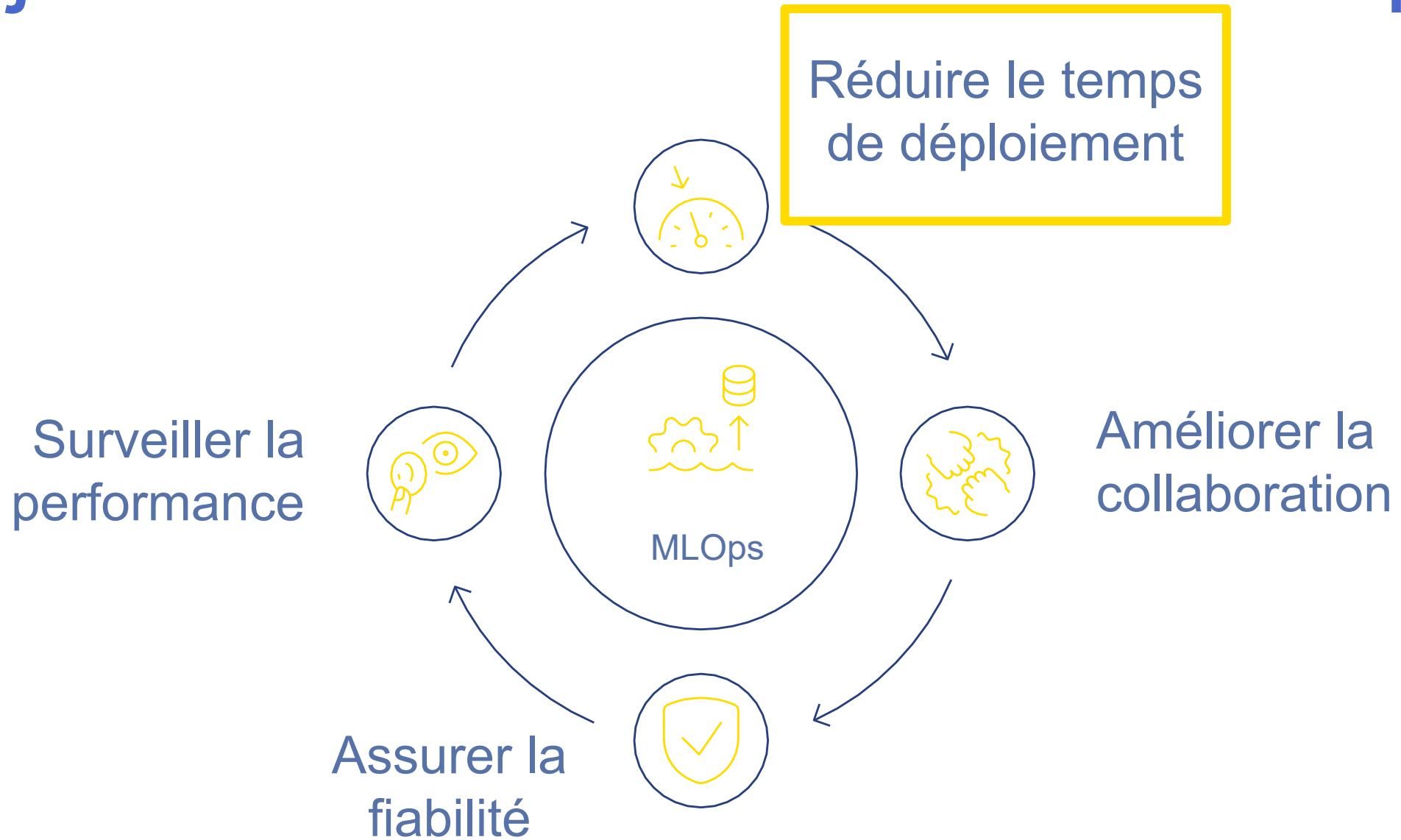
Modèles instables ou dégradant leurs performances sans mécanismes de contrôle et de maintenance.



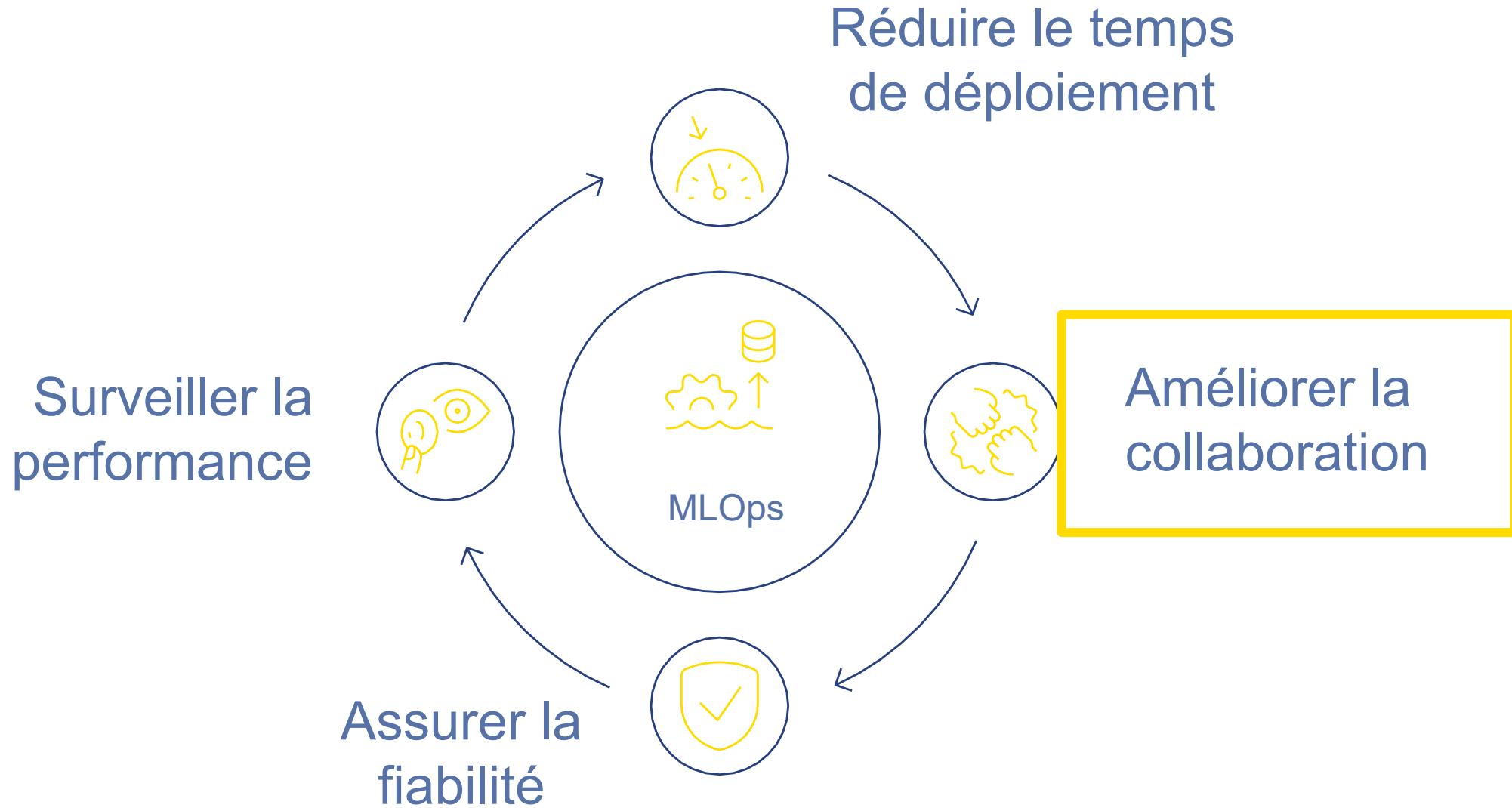
# Enjeux et bénéfices du MLOps ?



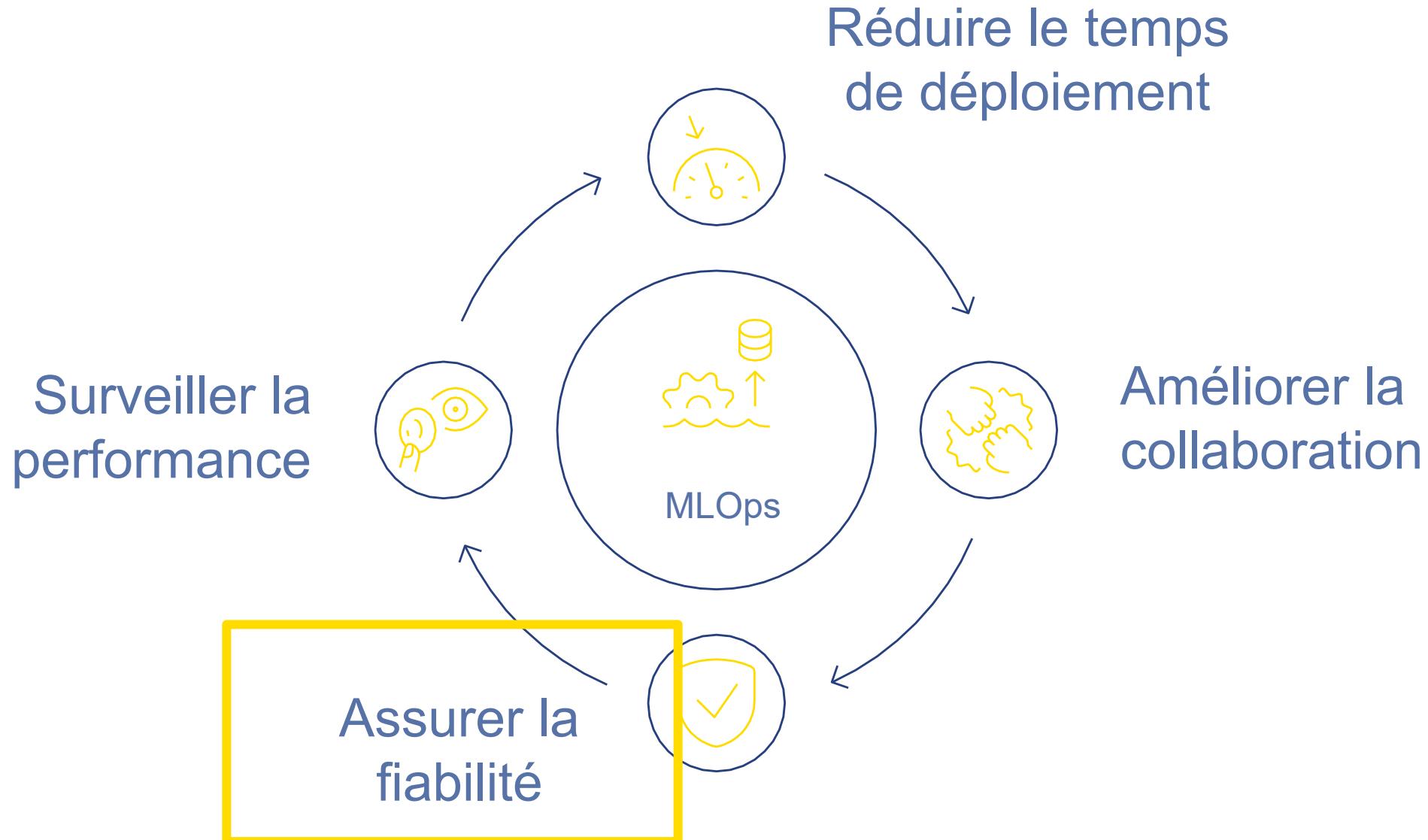
# Enjeux et bénéfices du MLOps ?



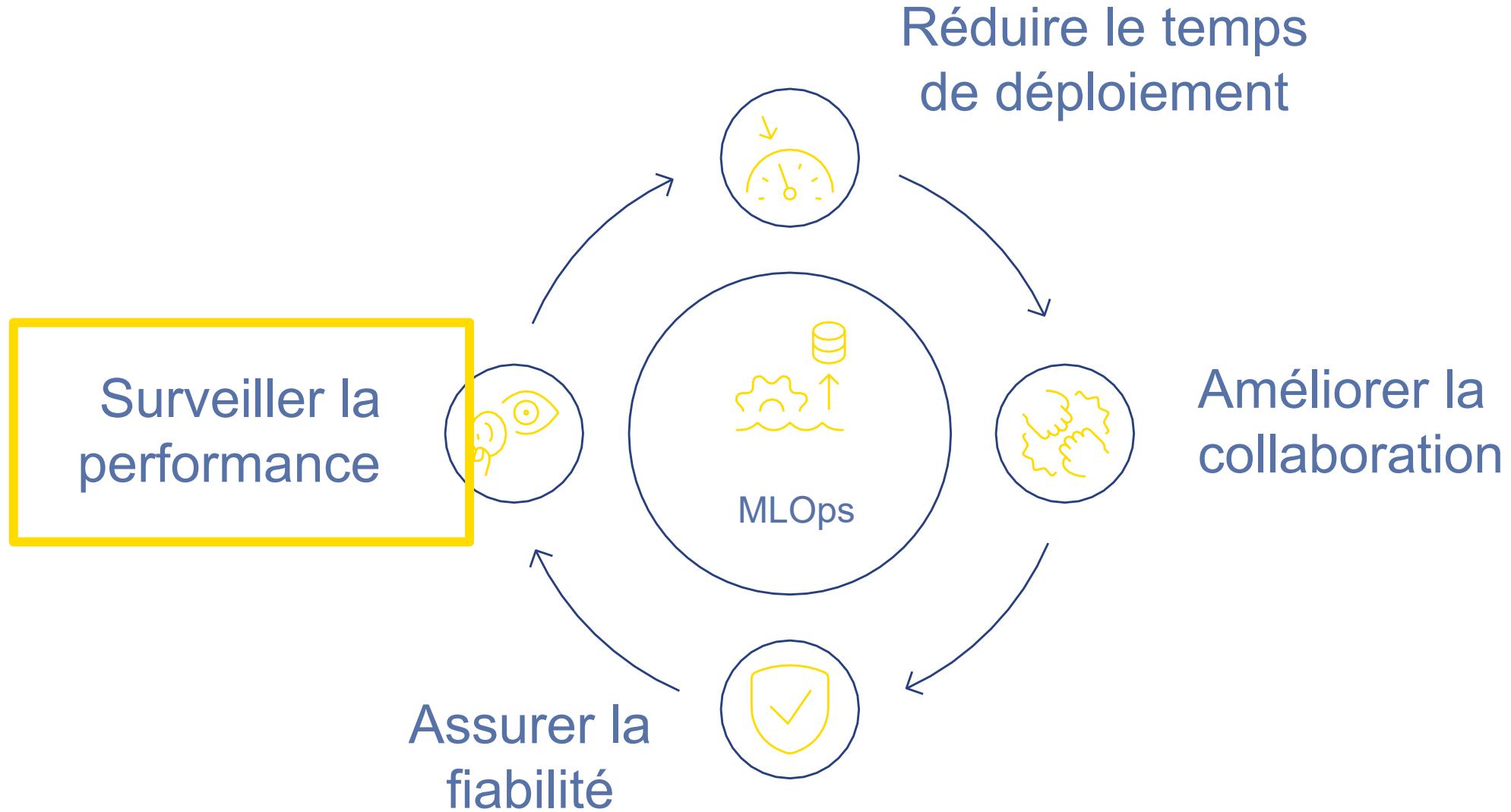
# Enjeux et bénéfices du MLOps ?



# Enjeux et bénéfices du MLOps ?



# Enjeux et bénéfices du MLOps ?



# En résumé

- MLOps = Process + Outils
- Complémentaire à la Data Science
- Objectif = Industrialiser, fiabiliser et sécuriser

# Expliquer le cycle de vie ML



Arthur Cartel Foahom Gouabou

Data Scientist & Chercheur en IA | Formateur IT

# Plan

- § Les grandes étapes du cycle de vie
- § Pourquoi structurer le cycle de vie ?

# Les grandes étapes du cycle de vie



- 01 Définition du problème
- 02 Collecte et préparation des données
- 03 Entraînement du modèle
- 04 Évaluation
- 05 Déploiement
- 06 Monitoring

# Les grandes étapes du cycle de vie



- 01 Définition du problème
- 02 Collecte et préparation des données
- 03 Entraînement du modèle
- 04 Évaluation
- 05 Déploiement
- 06 Monitoring

# Les grandes étapes du cycle de vie



- 01 Définition du problème
- 02 Collecte et préparation des données
- 03 Entraînement du modèle
- 04 Évaluation
- 05 Déploiement
- 06 Monitoring

# Les grandes étapes du cycle de vie



- 01 Définition du problème
- 02 Collecte et préparation des données
- 03 Entraînement du modèle
- 04 Évaluation
- 05 Déploiement
- 06 Monitoring

# Les grandes étapes du cycle de vie



- 01 Définition du problème
- 02 Collecte et préparation des données
- 03 Entraînement du modèle
- 04 Évaluation
- 05 Déploiement
- 06 Monitoring

# Les grandes étapes du cycle de vie



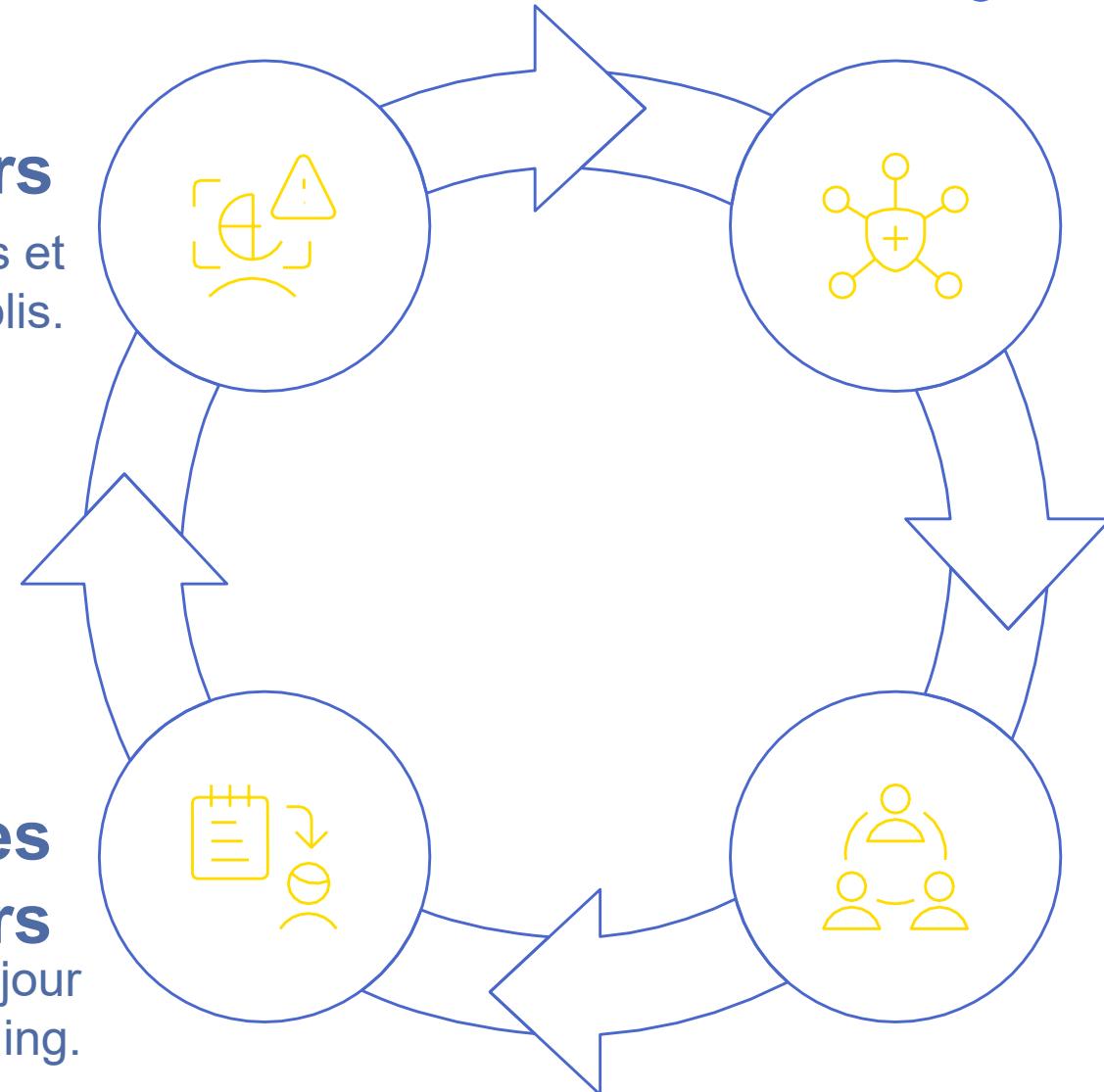
- 01 Définition du problème
- 02 Collecte et préparation des données
- 03 Entraînement du modèle
- 04 Évaluation
- 05 Déploiement
- 06 Monitoring

# Pourquoi structure le cycle de vie ?

---

## Éviter les erreurs

Minimiser les erreurs et les oubli.



## Anticiper les besoins futurs

Préparer les mises à jour et le retraining.

## Assurer la fiabilité

Garantir la stabilité et la cohérence du projet.

## Faciliter la collaboration

Améliorer le travail d'équipe et la communication.

# En résumé

- Le cycle de vie d'un projet ML est itératif et non linéaire.
- Chaque étape est critique pour assurer la réussite du projet.
- Bien comprendre ce cycle est la première clé pour industrialiser ses projets ML.

# Installer l'environnement (python, VSCode, Github)



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Installer python



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Installer VSCode, windsurf (copilot IA), extensions clés



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Installer Git et configurer les variables globales



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Créer un compte github, créer une clé ssh et faire votre premier push



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Mettre en œuvre un processus d'ingestion de données



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Préparer un environnement Linux et créer un utilisateur pour un serveur SFTP



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Sécuriser un serveur SFTP en isolant les répertoires et en configurant SSH



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Tester un serveur SFTP avec FileZilla



**Arthur Cartel Foahom Gouabou**

Data Scientist & Chercheur en IA | Formateur IT

# Structurer efficacement les dossiers d'un projet de Machine Learning



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Objectifs

# Plan

§ Objectifs

§ Pourquoi structurer un projet ML ?

# Plan

§ Objectifs

§ Pourquoi structurer un projet ML ?

§ Structure typique d'un projet ML

# Plan

§ Objectifs

§ Pourquoi structurer un projet ML ?

§ Structure typique d'un projet ML

§ Bonnes pratiques à retenir

# Objectifs

- Comprendre l'importance de structurer son projet

# Objectifs

- Comprendre l'importance de structurer son projet
- Découvrir une arborescence standard et professionnelle

# Objectifs

- Comprendre l'importance de structurer son projet
- Découvrir une arborescence standard et professionnelle
- Préparer le projet pour l'intégration MLOps (MLflow, DVC, Docker...)

# Pourquoi structurer un projet ML

## ?

- Améliore la lisibilité et la collaboration

# Pourquoi structurer un projet ML

## ?

- Améliore la lisibilité et la collaboration
- Facilite l'automatisation (pipelines)

# Pourquoi structurer un projet ML

## ?

- Améliore la lisibilité et la collaboration
- Facilite l'automatisation (pipelines)
- Permet la reproductibilité (versioning, logs)

# Pourquoi structurer un projet ML

## ?

- Améliore la lisibilité et la collaboration
- Facilite l'automatisation (pipelines)
- Permet la reproductibilité (versioning, logs)
- Prépare l'intégration d'outils comme DVC, MLflow, Docker

# Structure typique d'un projet ML

```
Project/
|
|--data/                      # Données brutes et prétraitées
|--notebooks/                  # Explorations Jupyter
|--src/                        # Code source (data, features, model)
|--models/                     # Modèles sauvegardés
|--pipelines/                  # Scripts de pipeline ou DVC
|--config/                     # Fichiers YAML ou JSON de configuration
|--tests/                       # Test unitaires
|--requirements.txt  # Dépendances
|--README.md                   # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|---data/          # Données brutes et prétraitées
|---notebooks/    # Explorations Jupyter
|---src/           # Code source (data, features, model)
|---models/        # Modèles sauvegardés
|---pipelines/    # Scripts de pipeline ou DVC
|---config/        # Fichiers YAML ou JSON de configuration
|---tests/          # Test unitaires
|---requirements.txt # Dépendances
|---README.md      # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/           # Données brutes et prétraitées
|   |--notebooks/  # Explorations Jupyter
|   |--src/         # Code source (data, features, model)
|   |--models/      # Modèles sauvegardés
|   |--pipelines/   # Scripts de pipeline ou DVC
|   |--config/      # Fichiers YAML ou JSON de configuration
|   |--tests/        # Test unitaires
|   |--requirements.txt  # Dépendances
|   |--README.md    # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/           # Données brutes et prétraitées
|--notebooks/      # Explorations Jupyter
|   |--src/          # Code source (data, features, model)
|   |--models/        # Modèles sauvegardés
|   |--pipelines/     # Scripts de pipeline ou DVC
|   |--config/         # Fichiers YAML ou JSON de configuration
|   |--tests/          # Test unitaires
|   |--requirements.txt # Dépendances
|   |--README.md       # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/                      # Données brutes et prétraitées
|--notebooks/                  # Explorations Jupyter
|--src/                        # Code source (data, features, model)
|   |--models/                  # Modèles sauvegardés
|   |--pipelines/               # Scripts de pipeline ou DVC
|   |--config/                  # Fichiers YAML ou JSON de configuration
|   |--tests/                   # Test unitaires
|   |--requirements.txt          # Dépendances
|   |--README.md                # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/                      # Données brutes et prétraitées
|--notebooks/                  # Explorations Jupyter
|--src/                        # Code source (data, features, model)
|--models/                     # Modèles sauvegardés
--pipelines/                   # Scripts de pipeline ou DVC
--config/                      # Fichiers YAML ou JSON de configuration
|--tests/                       # Test unitaires
|--requirements.txt  # Dépendances
|--README.md                    # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/                      # Données brutes et prétraitées
|--notebooks/                  # Explorations Jupyter
|--src/                        # Code source (data, features, model)
|--models/                     # Modèles sauvegardés
|--pipelines/                  # Scripts de pipeline ou DVC
|--config/                     # Fichiers YAML ou JSON de configuration
|--tests/                       # Tests unitaires
|--requirements.txt  # Dépendances
|--README.md                   # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/                      # Données brutes et prétraitées
|--notebooks/                  # Explorations Jupyter
|--src/                        # Code source (data, features, model)
|--models/                     # Modèles sauvegardés
|--pipelines/                  # Scripts de pipeline ou DVC
|--confia/                     # Fichiers YAML ou JSON de configuration
|--tests/                       # Test unitaires
|--requirements.txt  # Dépendances
|--README.md                   # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/                      # Données brutes et prétraitées
|--notebooks/                  # Explorations Jupyter
|--src/                        # Code source (data, features, model)
|--models/                     # Modèles sauvegardés
|--pipelines/                  # Scripts de pipeline ou DVC
|--config/                     # Fichiers YAML ou JSON de configuration
|--tests/                       # Test unitaires
|--requirements.txt  # Dépendances
|--README.md                 # Présentation du projet
```

# Structure typique d'un projet ML

```
Project/
|
|--data/                      # Données brutes et prétraitées
|--notebooks/                  # Explorations Jupyter
|--src/                        # Code source (data, features, model)
|--models/                     # Modèles sauvegardés
|--pipelines/                  # Scripts de pipeline ou DVC
|--config/                     # Fichiers YAML ou JSON de configuration
|--tests/                       # Test unitaires
|--requirements.txt  # Dépendances
--README.md                    # Présentation du projet
```

# Bonnes pratiques à retenir

- Toujours séparer le code, les données et les notebooks
- Ne jamais versionner les données brutes dans Git
- Préparer l'organisation dès le début du projet
- Nommer les fichiers et dossiers de façon explicite

# Bonnes pratiques à retenir

- Toujours séparer le code, les données et les notebooks
- Ne jamais versionner les données brutes dans Git
- Préparer l'organisation dès le début du projet
- Nommer les fichiers et dossiers de façon explicite

# Bonnes pratiques à retenir

- Toujours séparer le code, les données et les notebooks
- Ne jamais versionner les données brutes dans Git
- Préparer l'organisation dès le début du projet
- Nommer les fichiers et dossiers de façon explicite

# Bonnes pratiques à retenir

- Toujours séparer le code, les données et les notebooks
- Ne jamais versionner les données brutes dans Git
- Préparer l'organisation dès le début du projet
- Nommer les fichiers et dossiers de façon explicite

# Bonnes pratiques à retenir

- Toujours séparer le code, les données et les notebooks
- Ne jamais versionner les données brutes dans Git
- Préparer l'organisation dès le début du projet
- Nommer les fichiers et dossiers de façon explicite

# Construire un pipeline ML modulaire



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Objectifs

# Plan

§ Objectifs

§ Pourquoi un pipeline modulaire en  
MLOps ?

# Plan

§ Objectifs

§ Pourquoi un pipeline modulaire en  
MLOps ?

§ Bonnes pratiques

# Plan

§ Objectifs

§ Pourquoi un pipeline modulaire en  
MLOps ?

§ Bonnes pratiques

§ Pipeline du projet churn

# Objectifs

# Objectifs

§ Comprendre l'intérêt de structurer  
un pipeline pour l'industrialisation

# Objectifs

- § Comprendre l'intérêt de structurer un pipeline pour l'industrialisation
- § Découvrir les bonnes pratiques de modularisation

# Objectifs

- § Comprendre l'intérêt de structurer un pipeline pour l'industrialisation
- § Découvrir les bonnes pratiques de modularisation
- § Explorer le pipeline MLOps appliqué au projet churn

# Pourquoi un pipeline modulaire en MLOps ?

# Pourquoi un pipeline modulaire en MLOps ?

§ Réutilisation des composants

# Pourquoi un pipeline modulaire en MLOps ?

§ Réutilisation des composants

§ Tests unitaires facilités

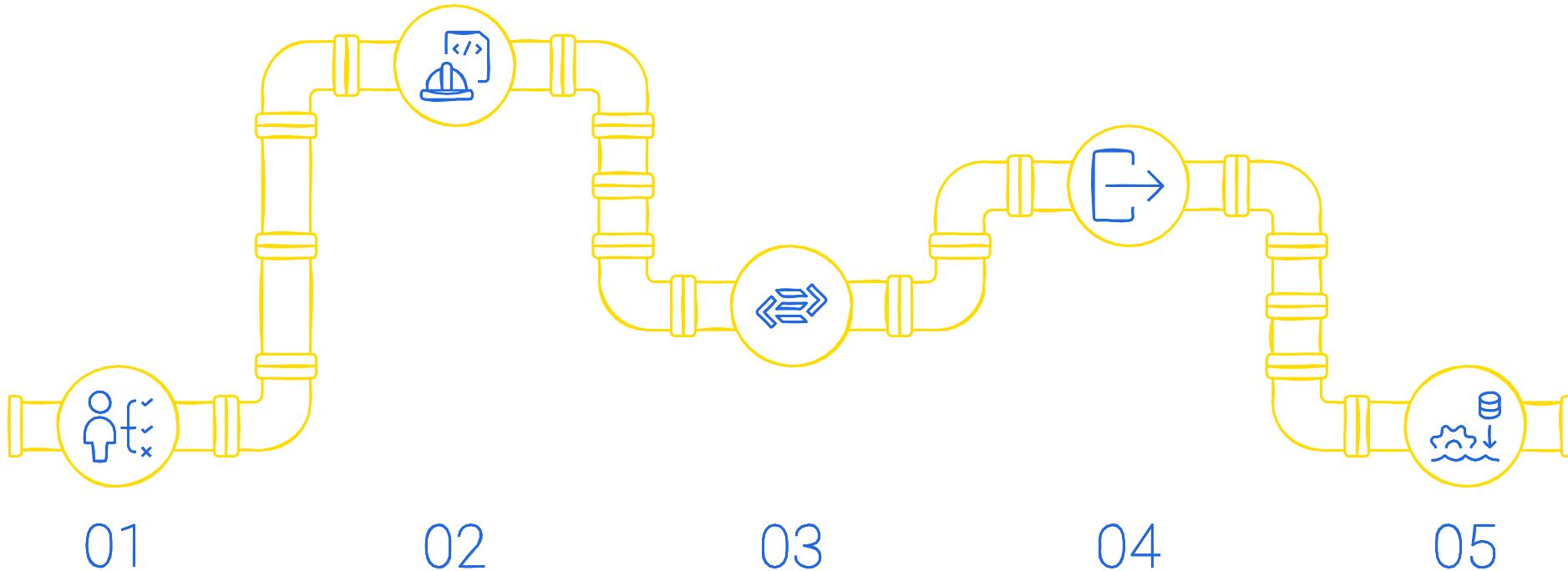
# Pourquoi un pipeline modulaire en MLOps ?

- § Réutilisation des composants
- § Tests unitaires facilités
- § Intégration avec des outils (DVC,  
MLflow, CI/CD)

# Pourquoi un pipeline modulaire en MLOps ?

- § Réutilisation des composants
- § Tests unitaires facilités
- § Intégration avec des outils (DVC,  
MLflow, CI/CD)
- § Orchestration (Airflow, Kubeflow...)

# Bonnes pratiques



## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

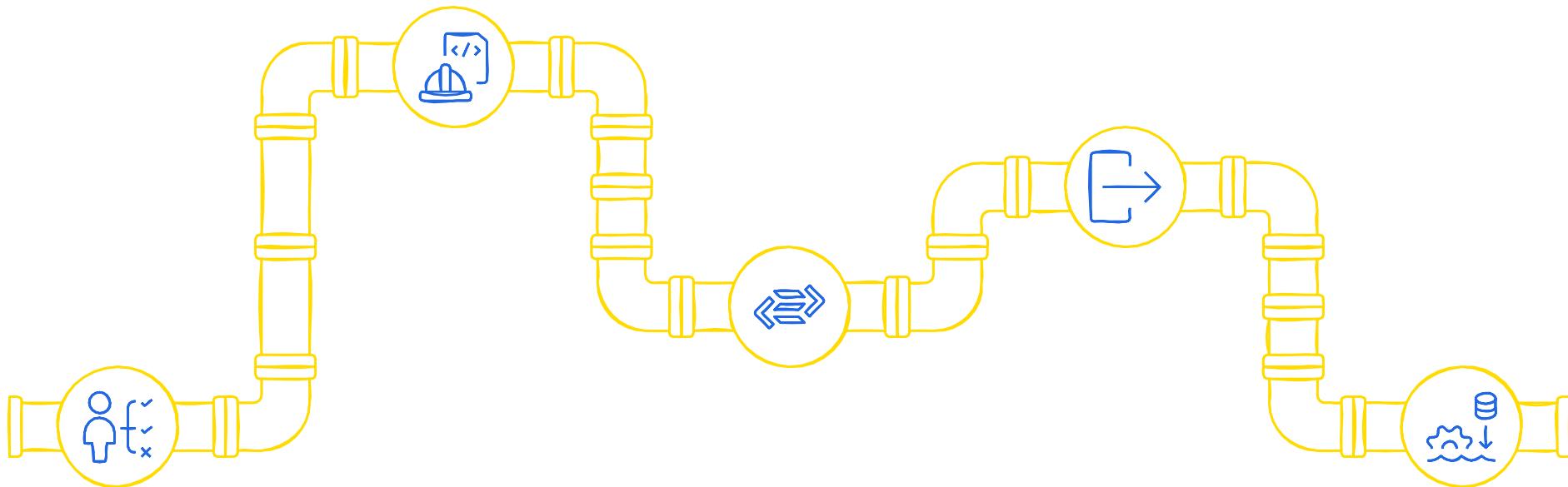
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

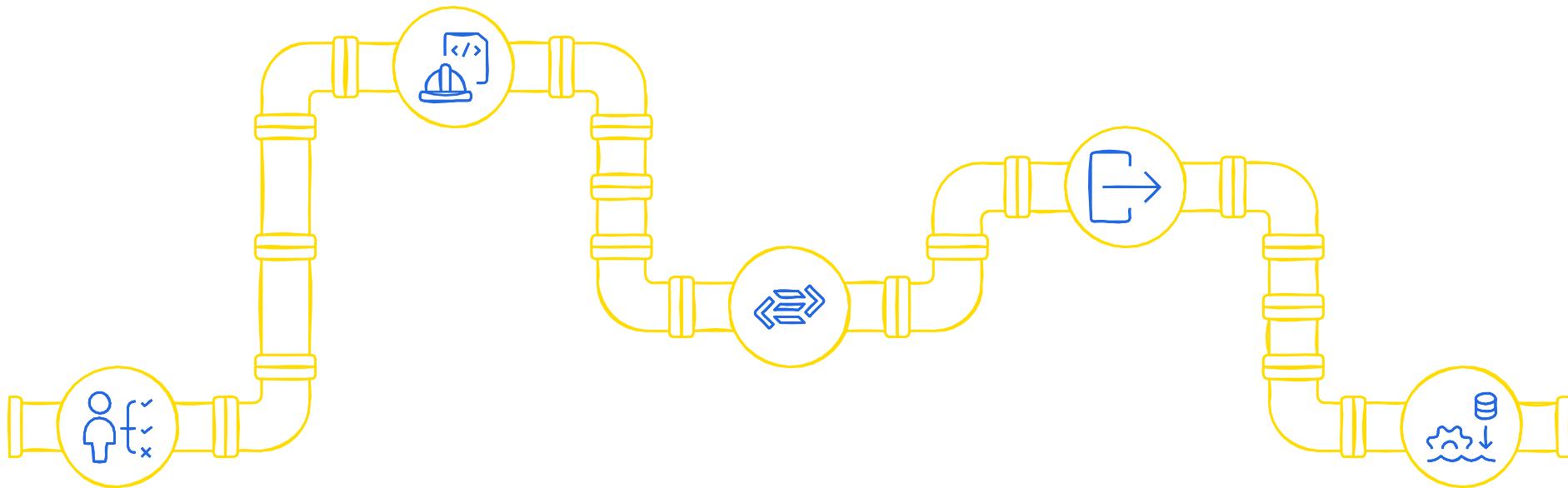
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

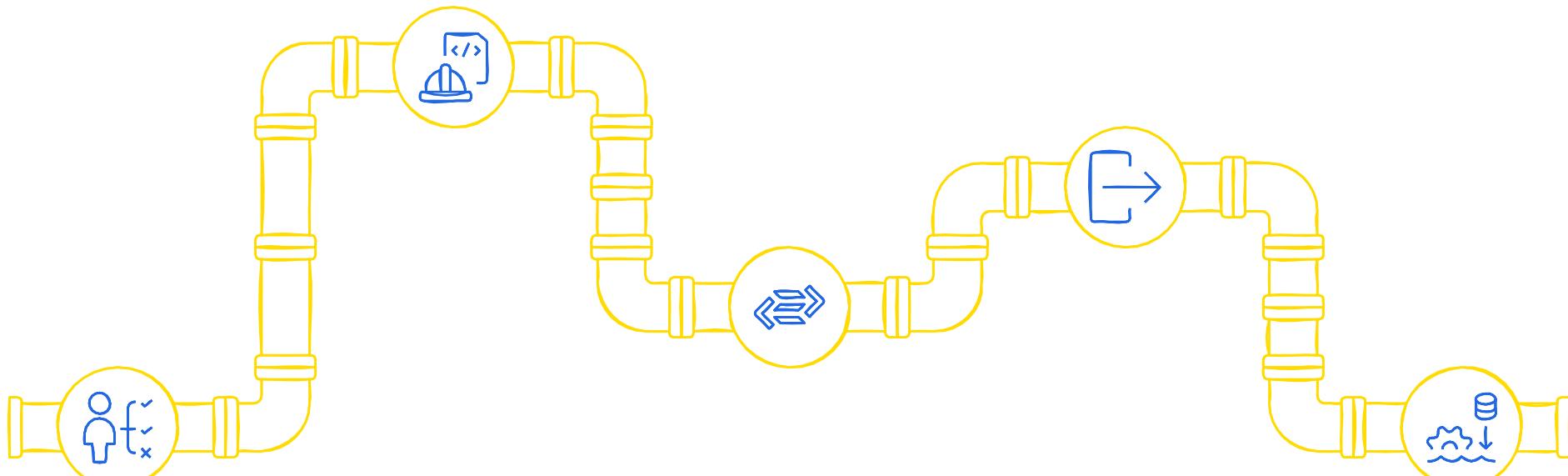
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

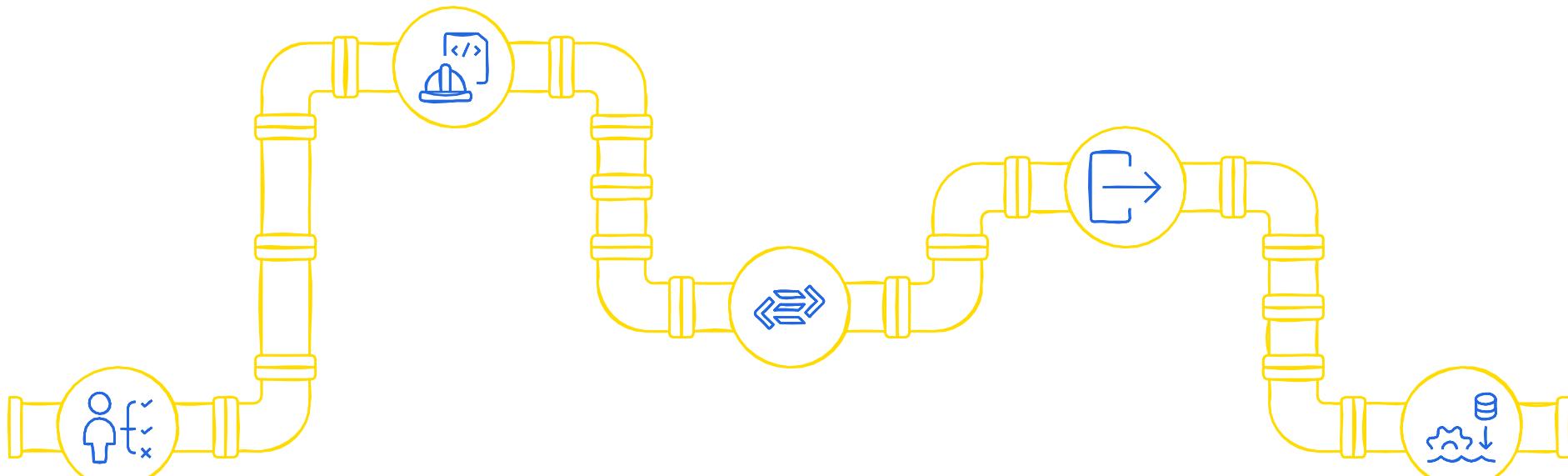
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

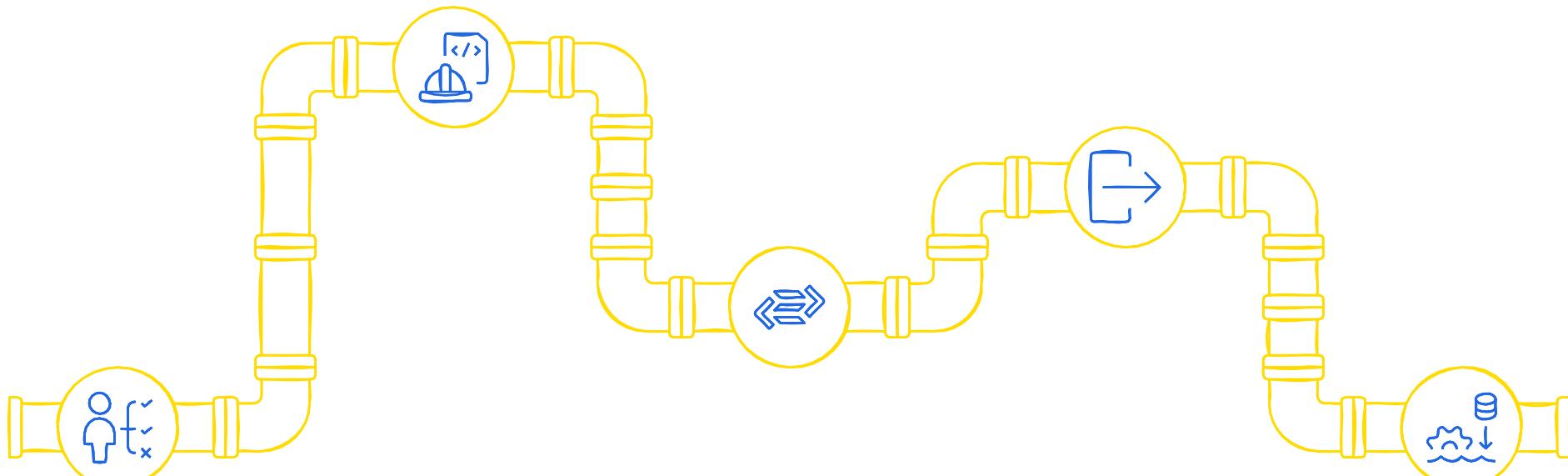
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

Utiliser un logger plutôt que des print()

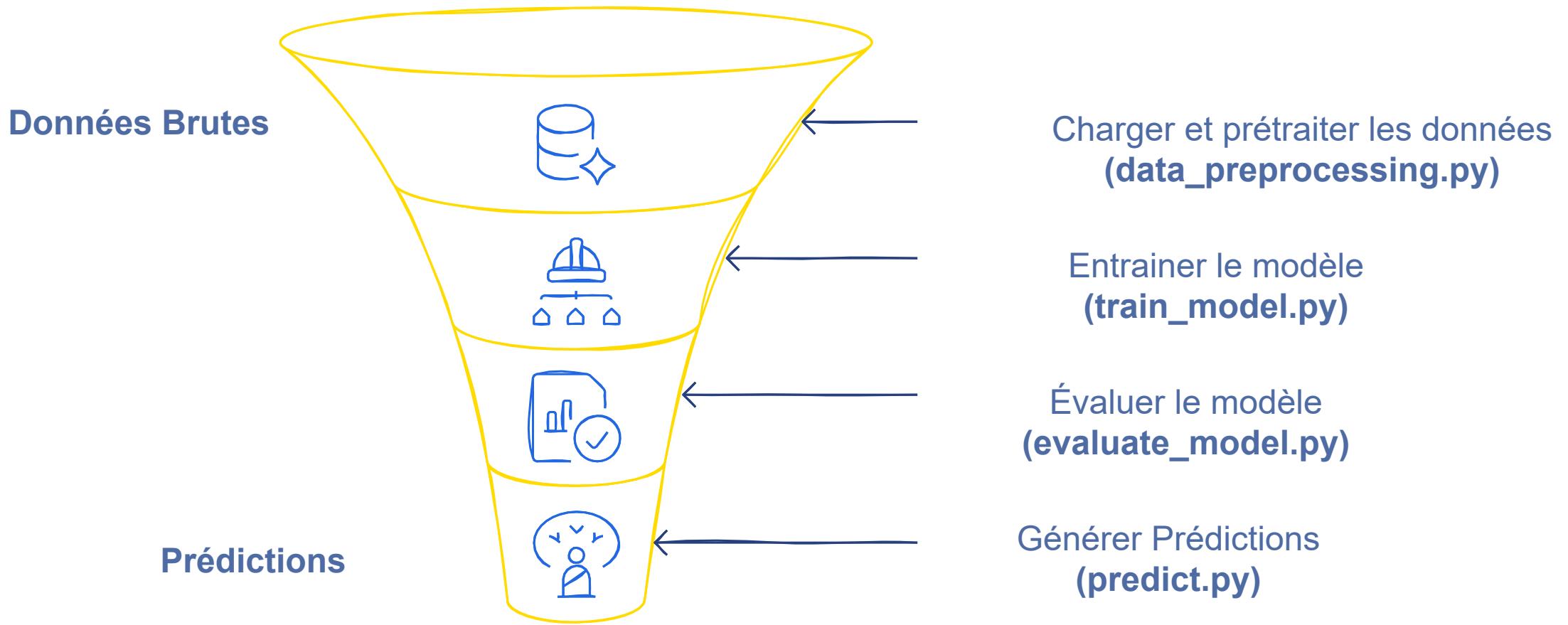
05

## Intégrer avec MLflow/DVC

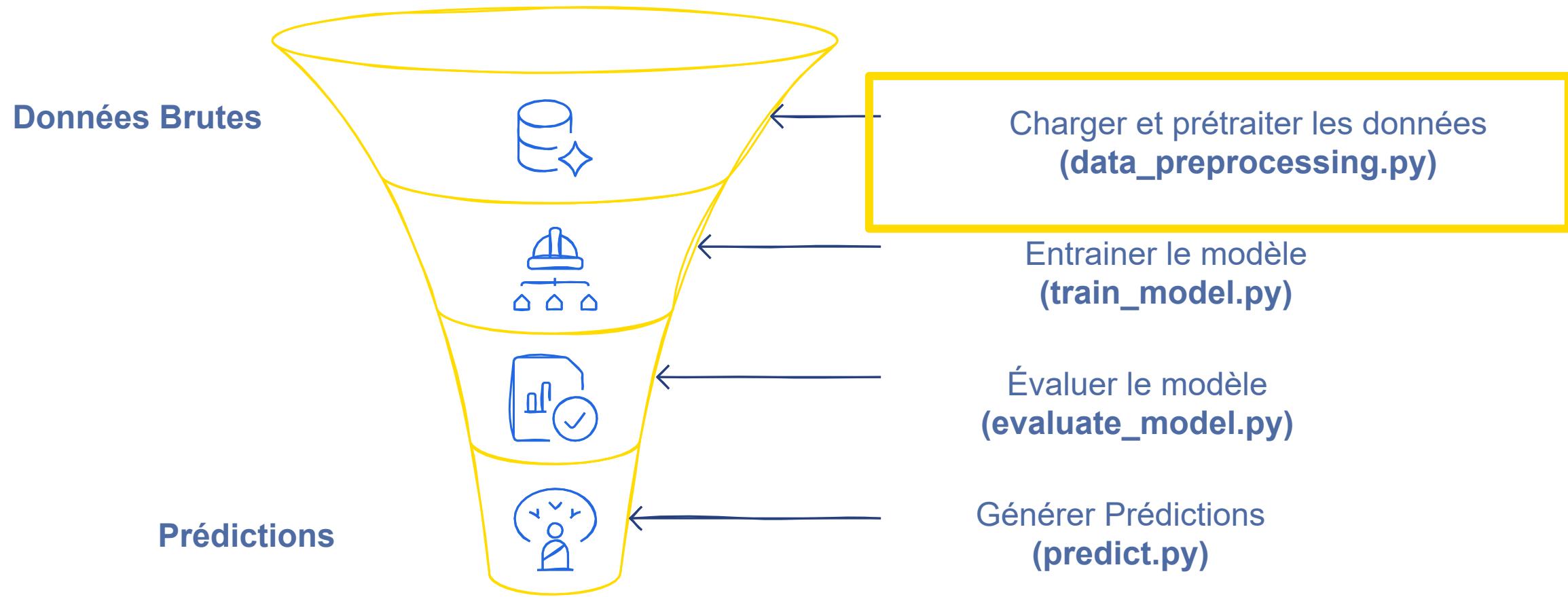
Préparer des points d'entrée pour les outils de suivi

# Pipeline du projet churn

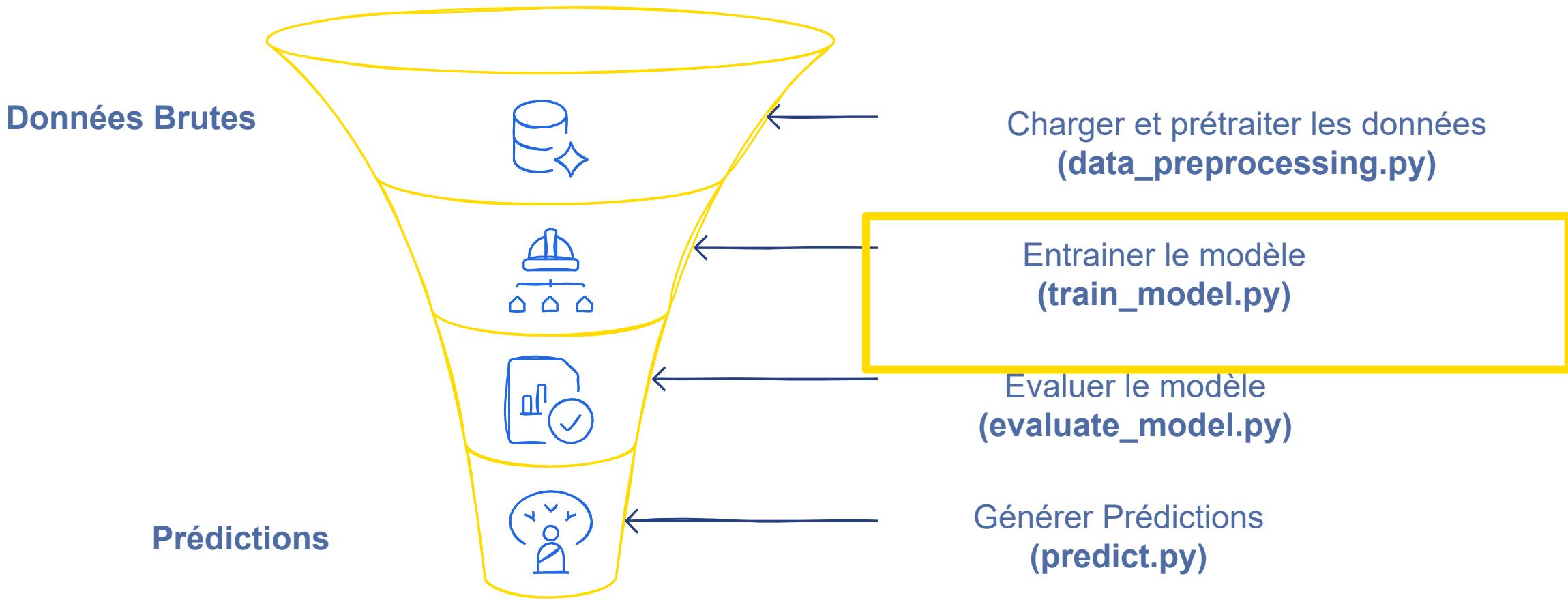
# Pipeline du projet churn



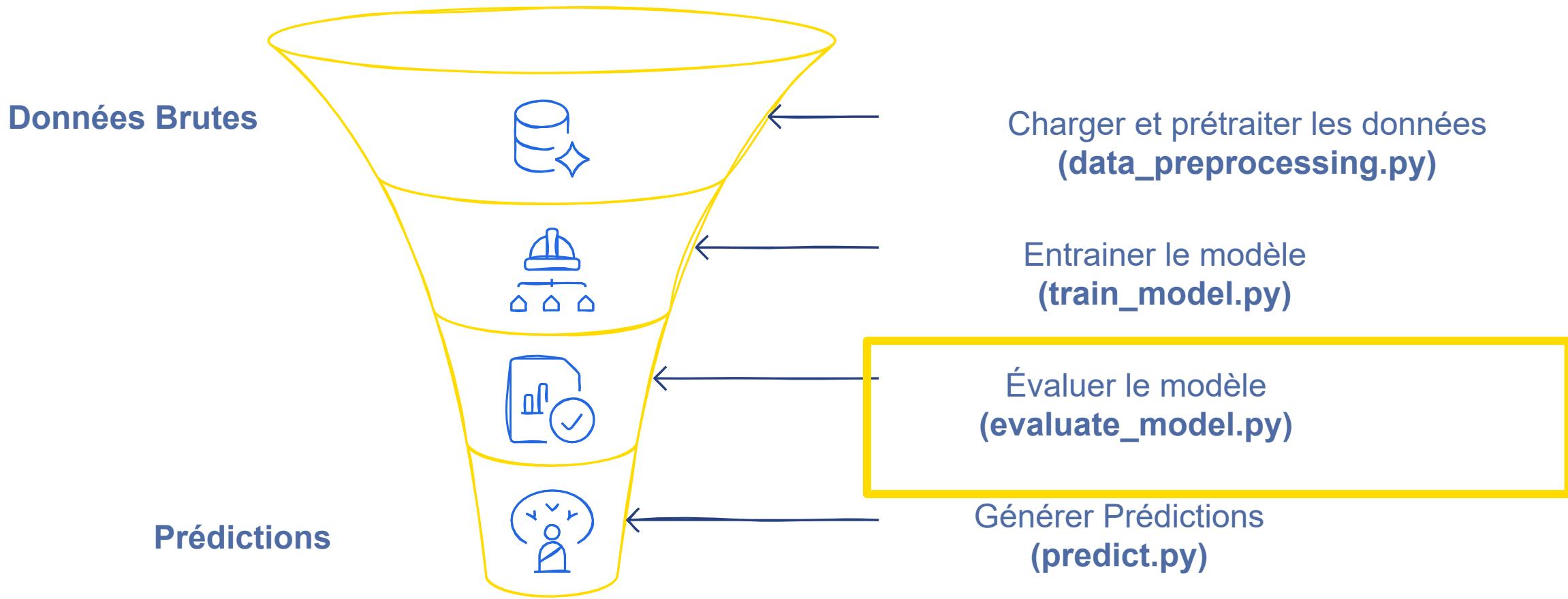
# Pipeline du projet churn



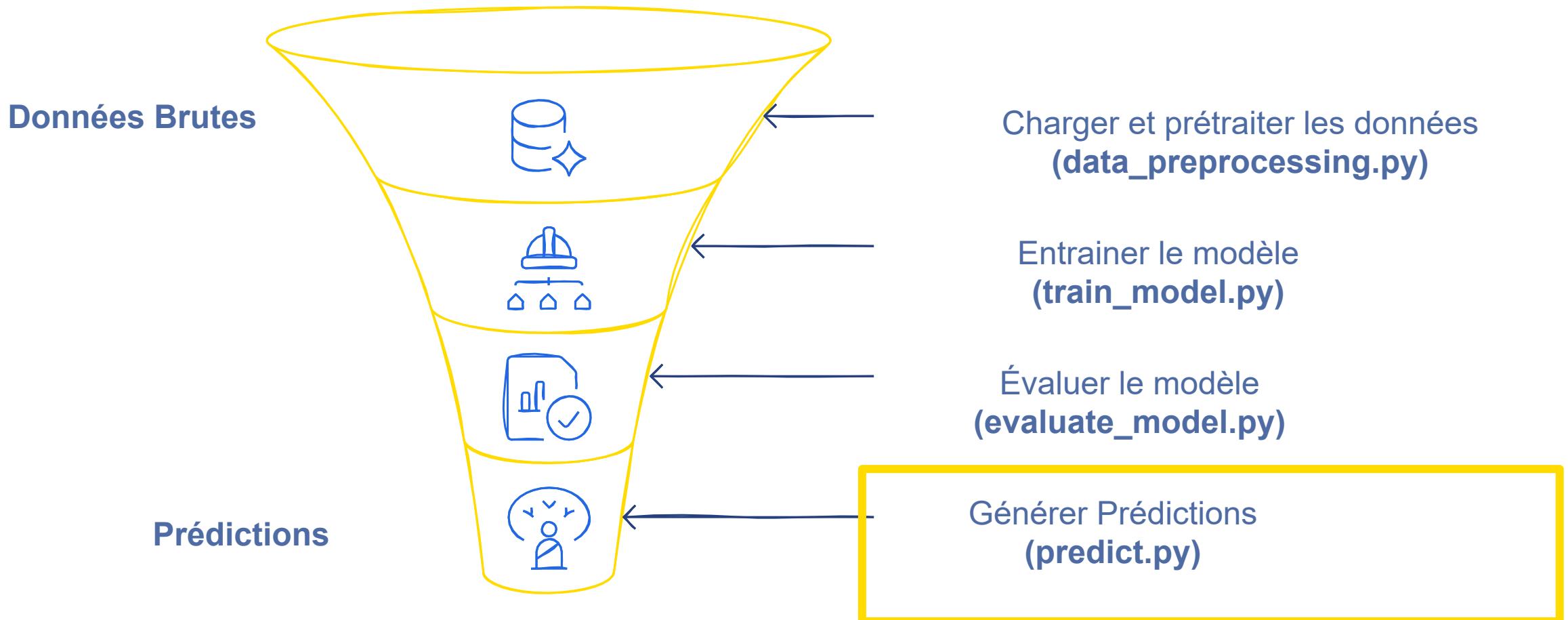
# Pipeline du projet churn



# Pipeline du projet churn



# Pipeline du projet churn



# Construire un pipeline ML modulaire



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Objectifs

# Plan

§ Objectifs

§ Pourquoi un pipeline modulaire en  
MLOps ?

# Plan

§ Objectifs

§ Pourquoi un pipeline modulaire en  
MLOps ?

§ Bonnes pratiques

# Plan

§ Objectifs

§ Pourquoi un pipeline modulaire en  
MLOps ?

§ Bonnes pratiques

§ Pipeline du projet churn

# Objectifs

# Objectifs

§ Comprendre l'intérêt de structurer  
un pipeline pour l'industrialisation

# Objectifs

- § Comprendre l'intérêt de structurer un pipeline pour l'industrialisation
- § Découvrir les bonnes pratiques de modularisation

# Objectifs

- § Comprendre l'intérêt de structurer un pipeline pour l'industrialisation
- § Découvrir les bonnes pratiques de modularisation
- § Explorer le pipeline MLOps appliqué au projet churn

# Pourquoi un pipeline modulaire en MLOps ?

# Pourquoi un pipeline modulaire en MLOps ?

§ Réutilisation des composants

# Pourquoi un pipeline modulaire en MLOps ?

§ Réutilisation des composants

§ Tests unitaires facilités

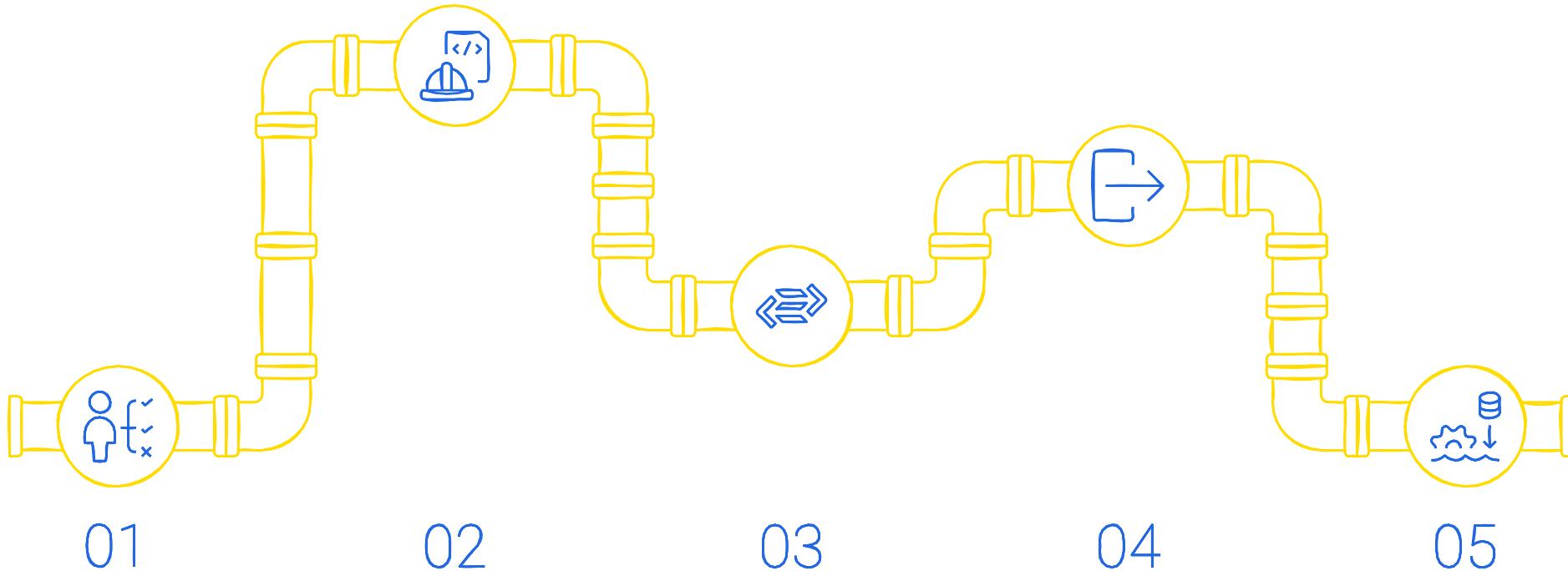
# Pourquoi un pipeline modulaire en MLOps ?

- § Réutilisation des composants
- § Tests unitaires facilités
- § Intégration avec des outils (DVC,  
MLflow, CI/CD)

# Pourquoi un pipeline modulaire en MLOps ?

- § Réutilisation des composants
- § Tests unitaires facilités
- § Intégration avec des outils (DVC,  
MLflow, CI/CD)
- § Orchestration (Airflow, Kubeflow...)

# Bonnes pratiques



## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

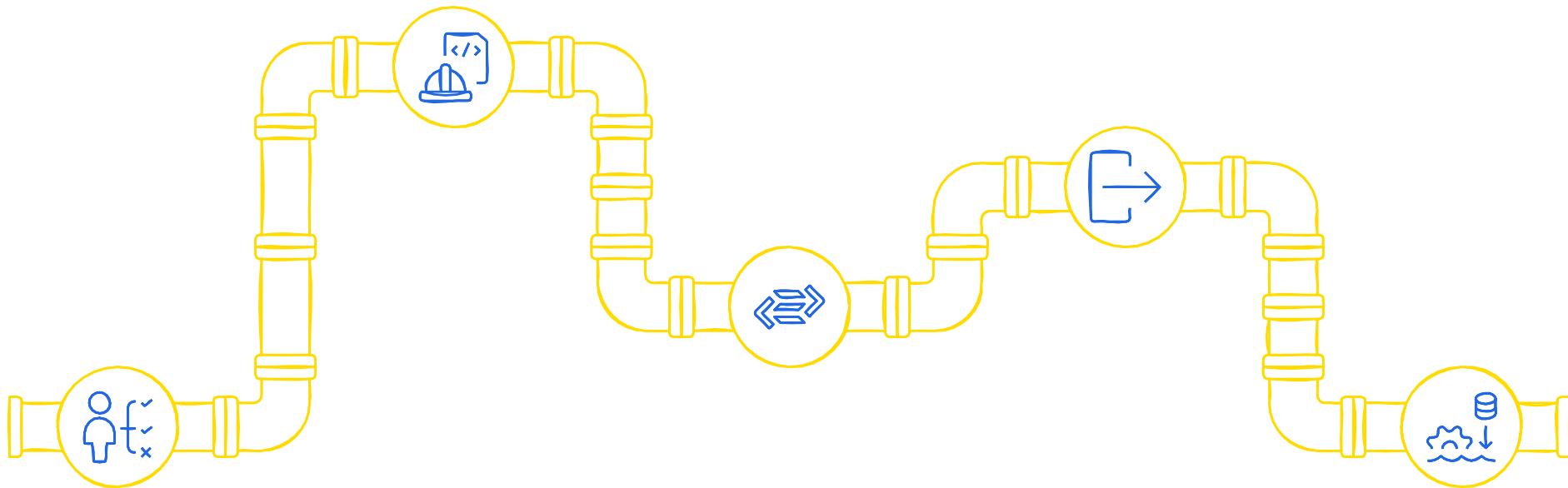
## Logger les sorties

Utiliser un logger plutôt que des print()

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

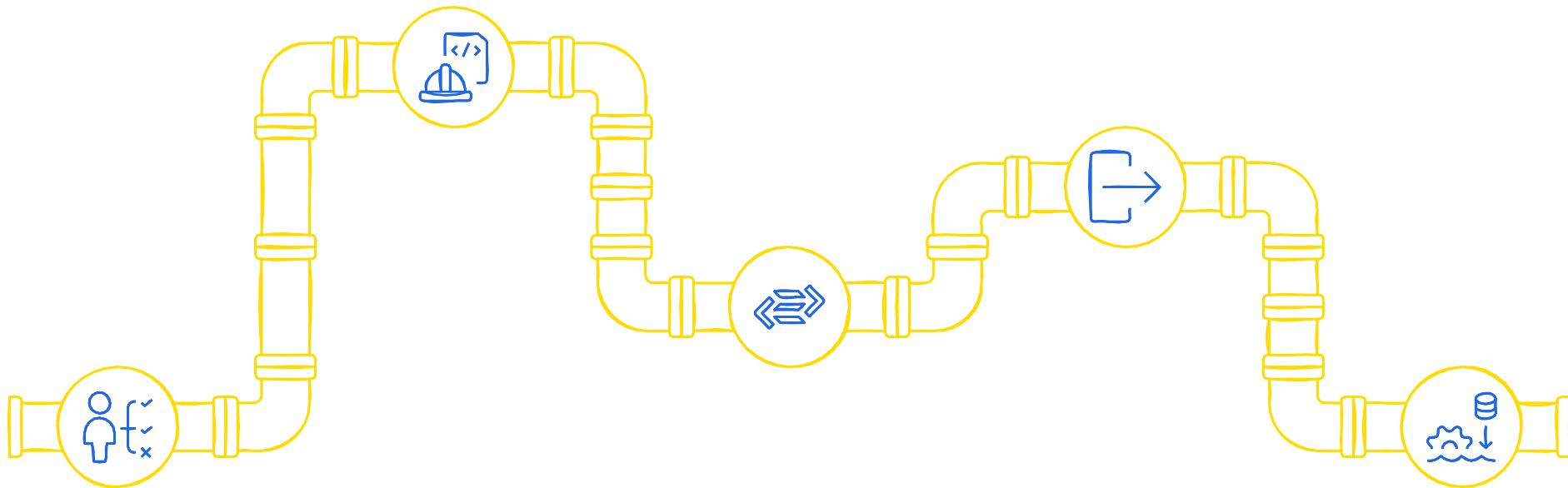
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

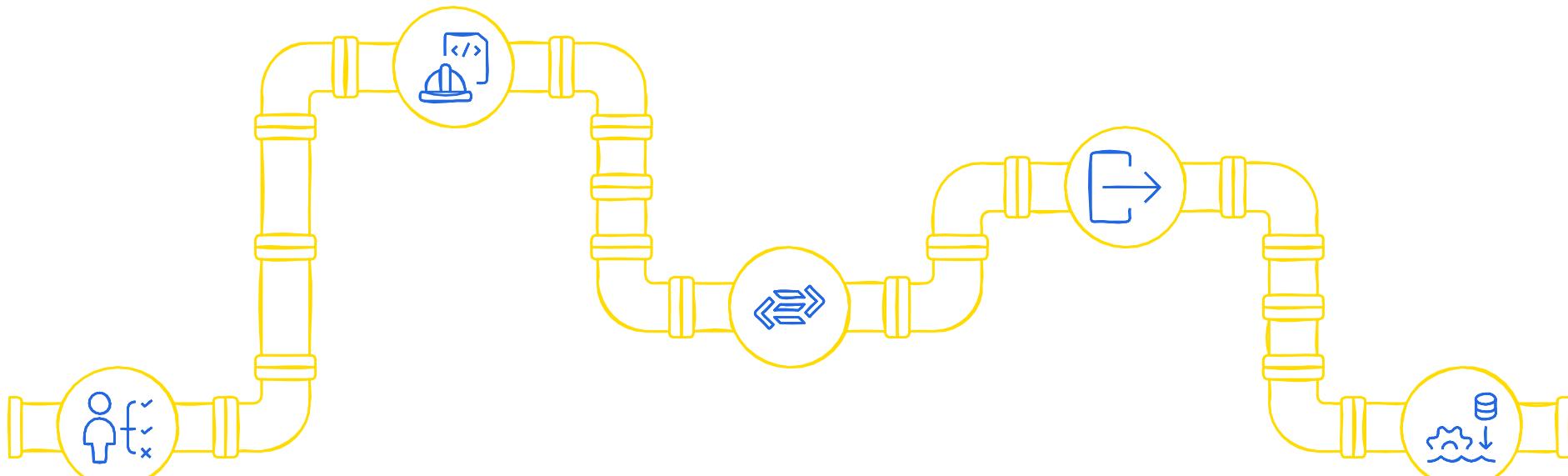
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

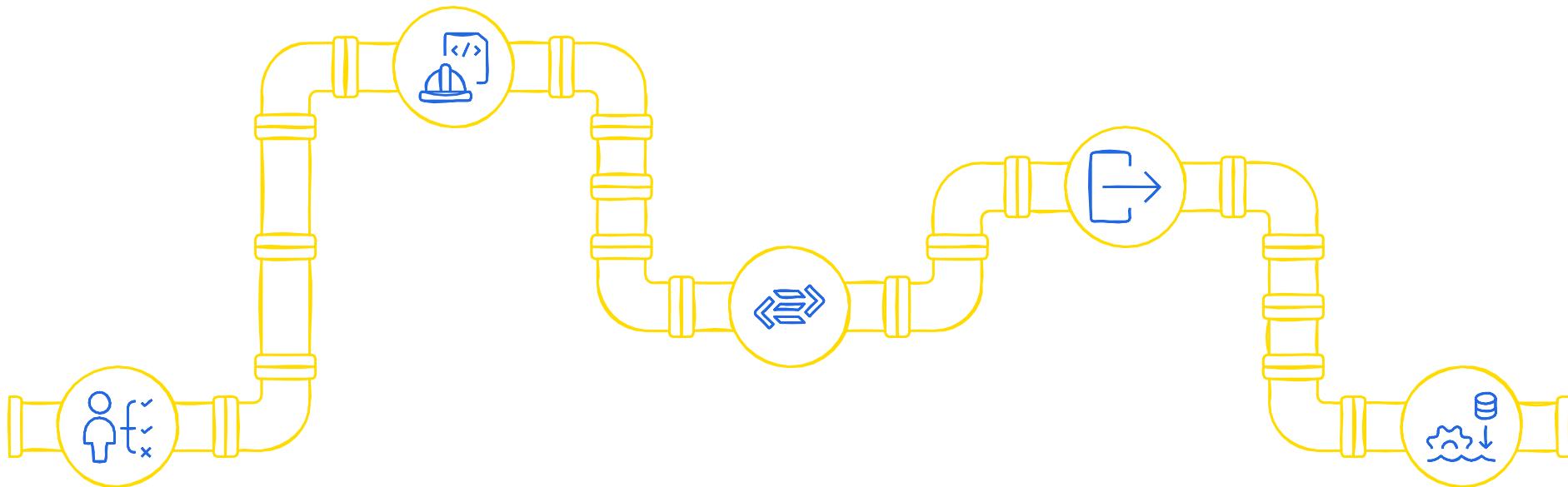
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

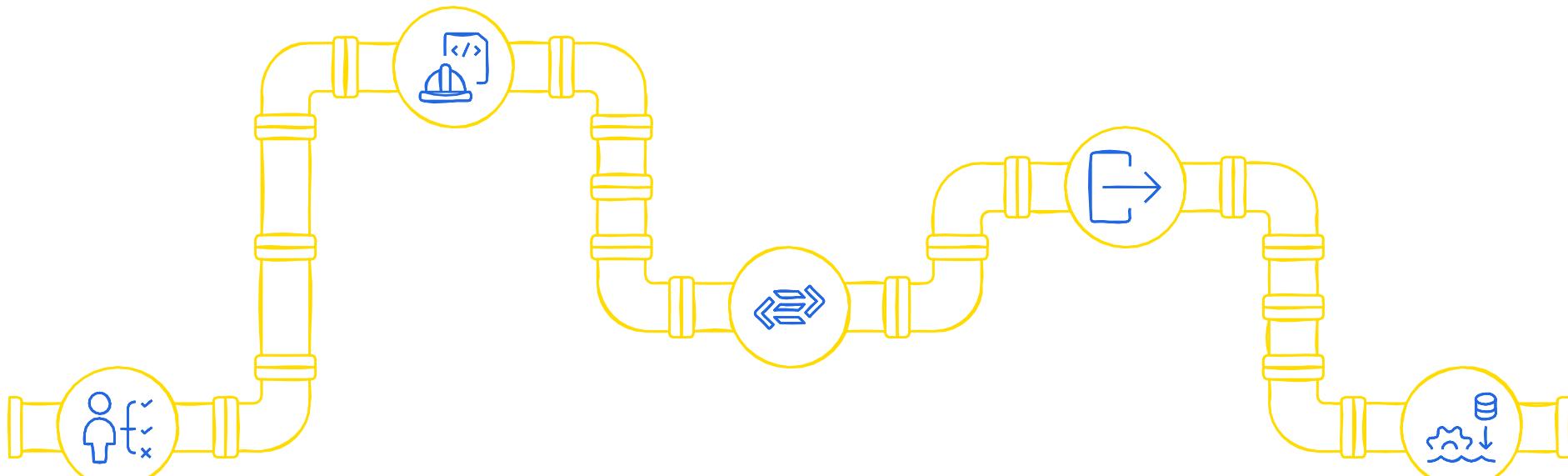
Utiliser un logger plutôt que des print()

05

## Intégrer avec MLflow/DVC

Préparer des points d'entrée pour les outils de suivi

# Bonnes pratiques



01

## Définir une tâche par script

Chaque script doit se concentrer sur une seule tâche

02

## Éviter le hardcoding

Utiliser des fichiers de configuration pour la flexibilité

03

## Scripts exécutables

Assurer que les scripts peuvent être exécutés via la ligne de commande

04

## Logger les sorties

Utiliser un logger plutôt que des print()

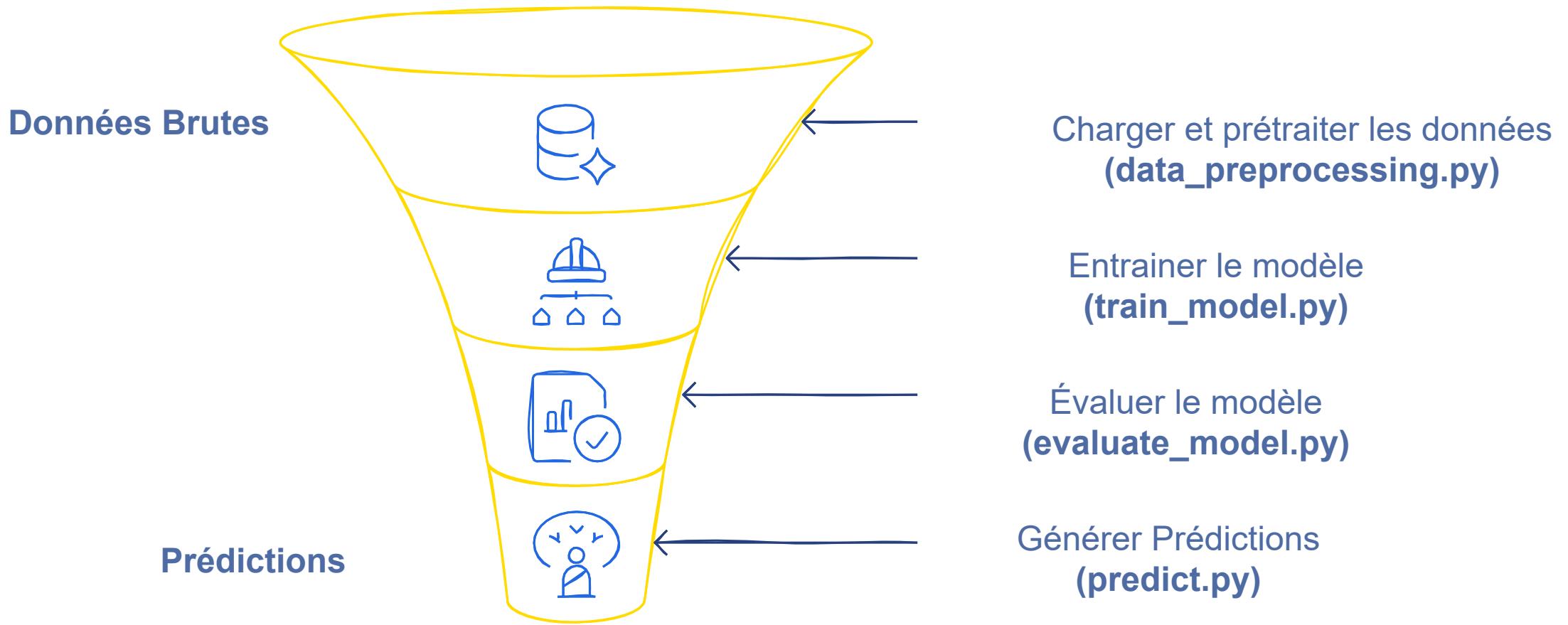
05

## Intégrer avec MLflow/DVC

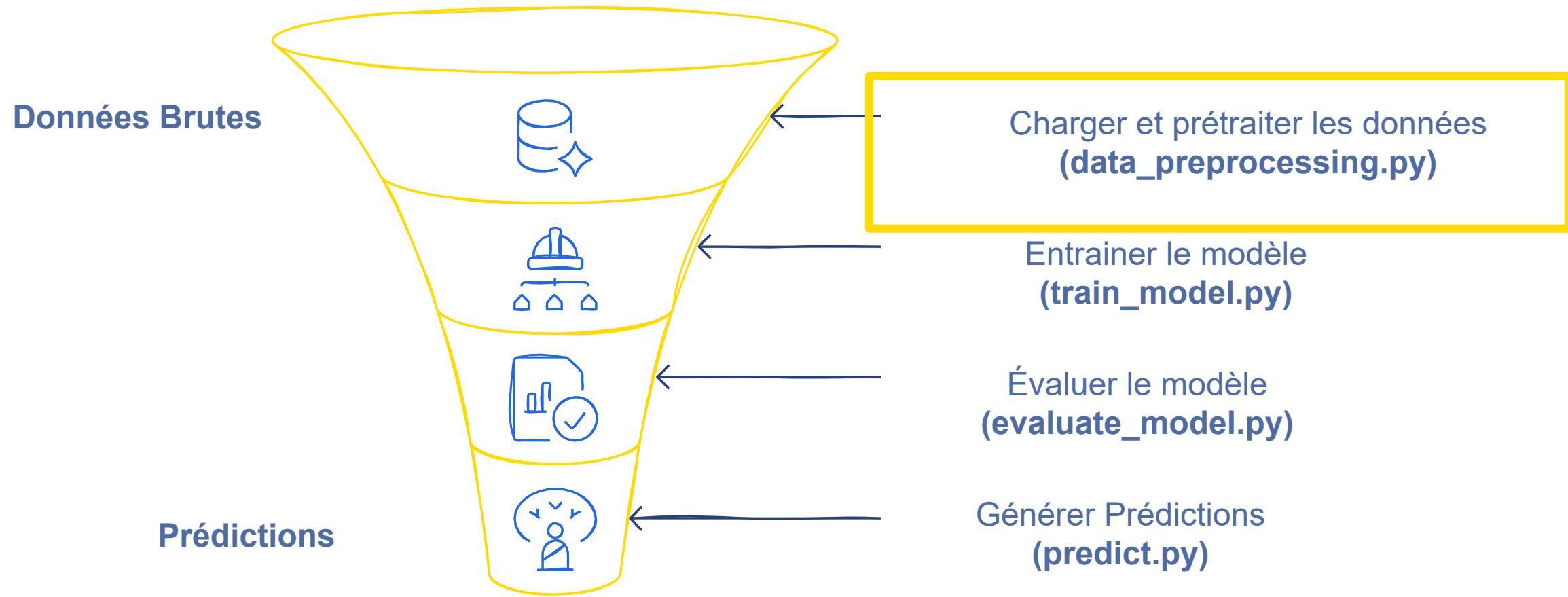
Préparer des points d'entrée pour les outils de suivi

# Pipeline du projet churn

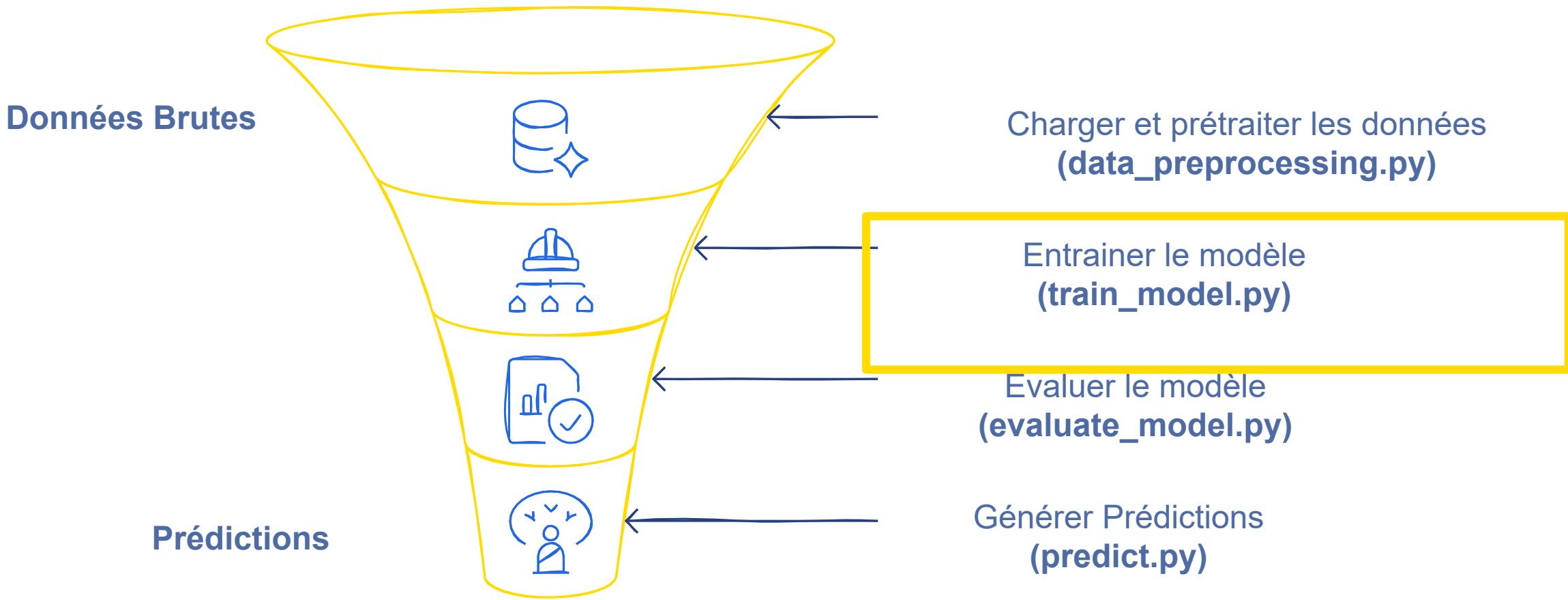
# Pipeline du projet churn



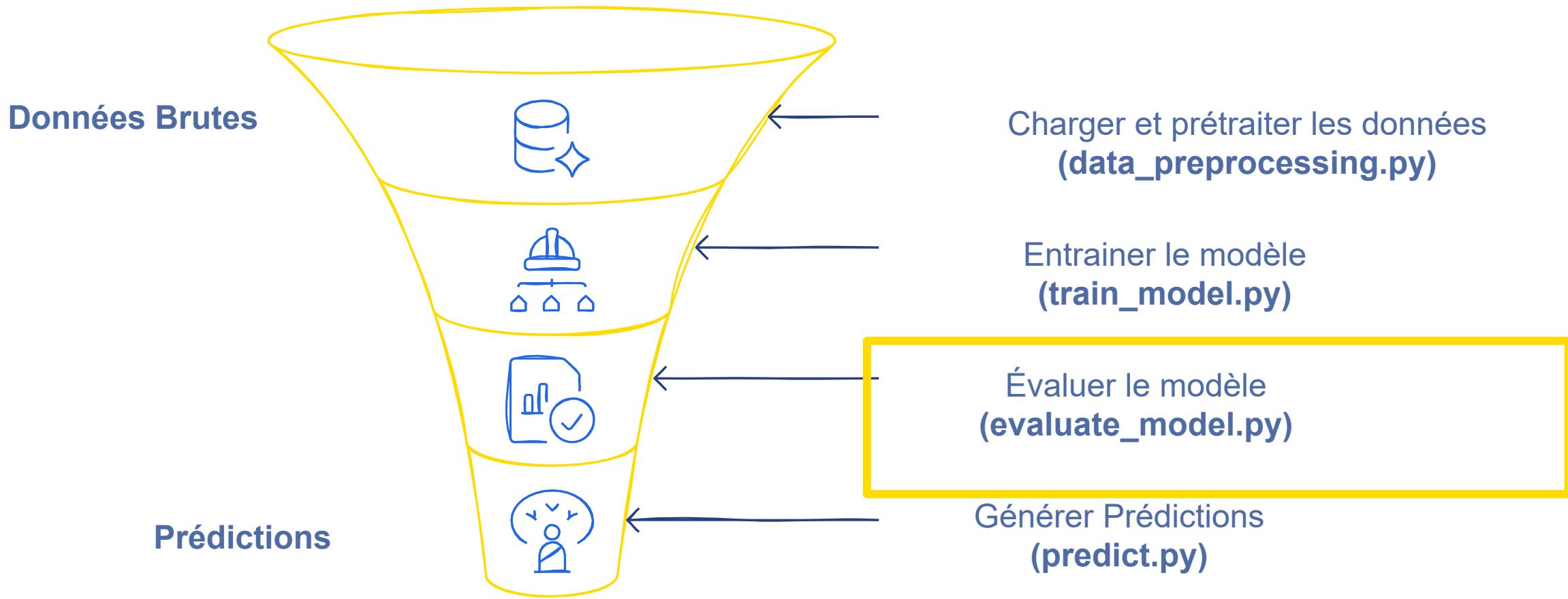
# Pipeline du projet churn



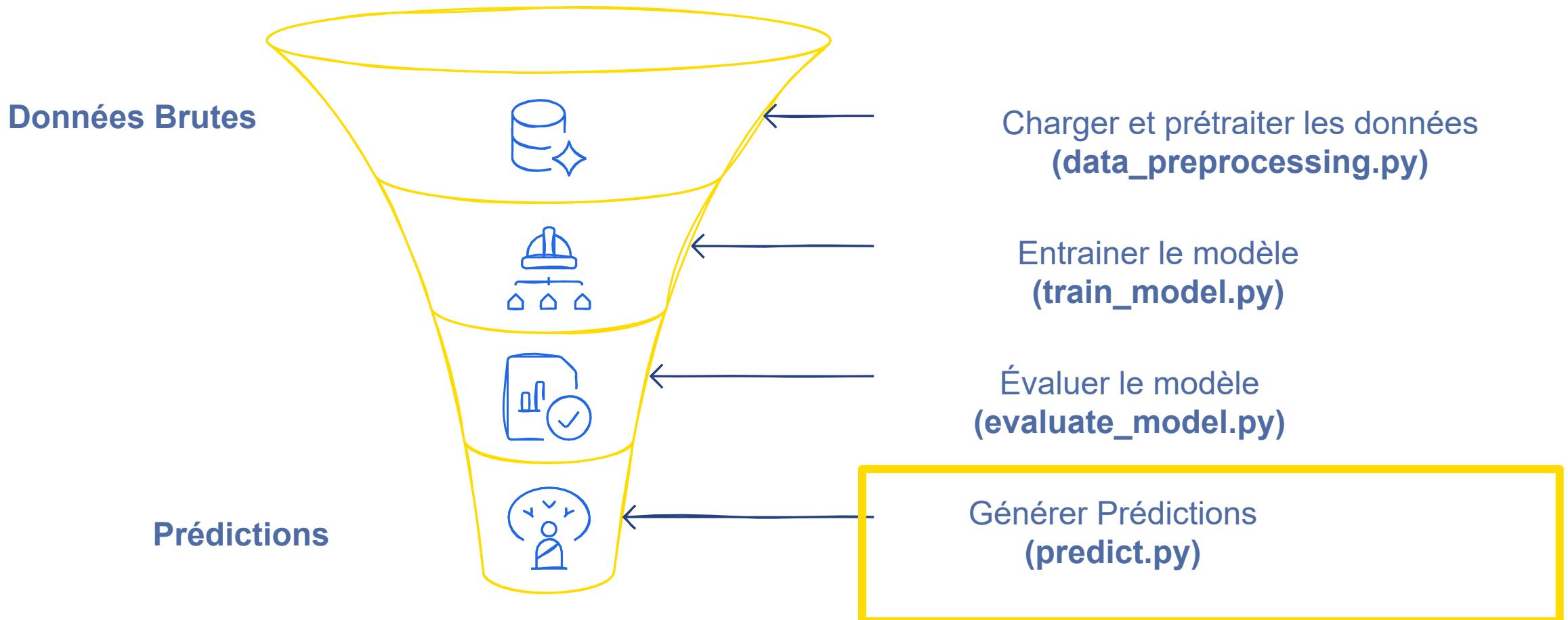
# Pipeline du projet churn



# Pipeline du projet churn



# Pipeline du projet churn



# Découvrir le rôle de MLflow en MLOps



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Introduction à MLflow

# Plan

§ Introduction à Mlflow

§ Présentation de ses 4 modules

# Plan

§ Introduction à Mlflow

§ Présentation de ses 4 modules

§ Avantages pour un projet en entreprise

# Plan

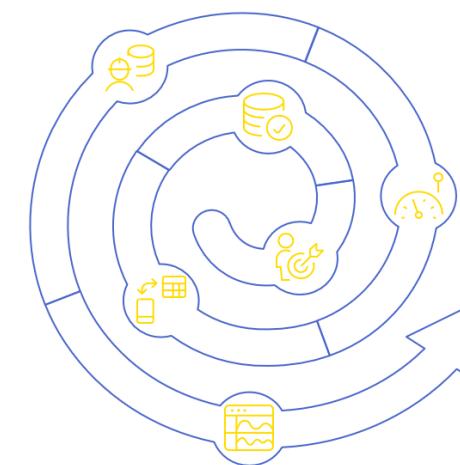
- § Introduction à Mlflow
- § Présentation de ses 4 modules
- § Avantages pour un projet en entreprise
- § Cas d'application : prédiction du churn

# Qu'est-ce que MLflow ?

**MLflow est un outil open-source développé par Databricks**, conçu pour simplifier la gestion du cycle de vie des modèles de machine learning.



## Les grandes étapes du cycle de vie



- 01 Définition du problème
- 02 Collecte et préparation des données
- 03 Entraînement du modèle
- 04 Évaluation
- 05 Déploiement
- 06 Monitoring

# Qu'est-ce que MLflow ?

- Permet de suivre, reproduire, packager et déployer les modèles ML

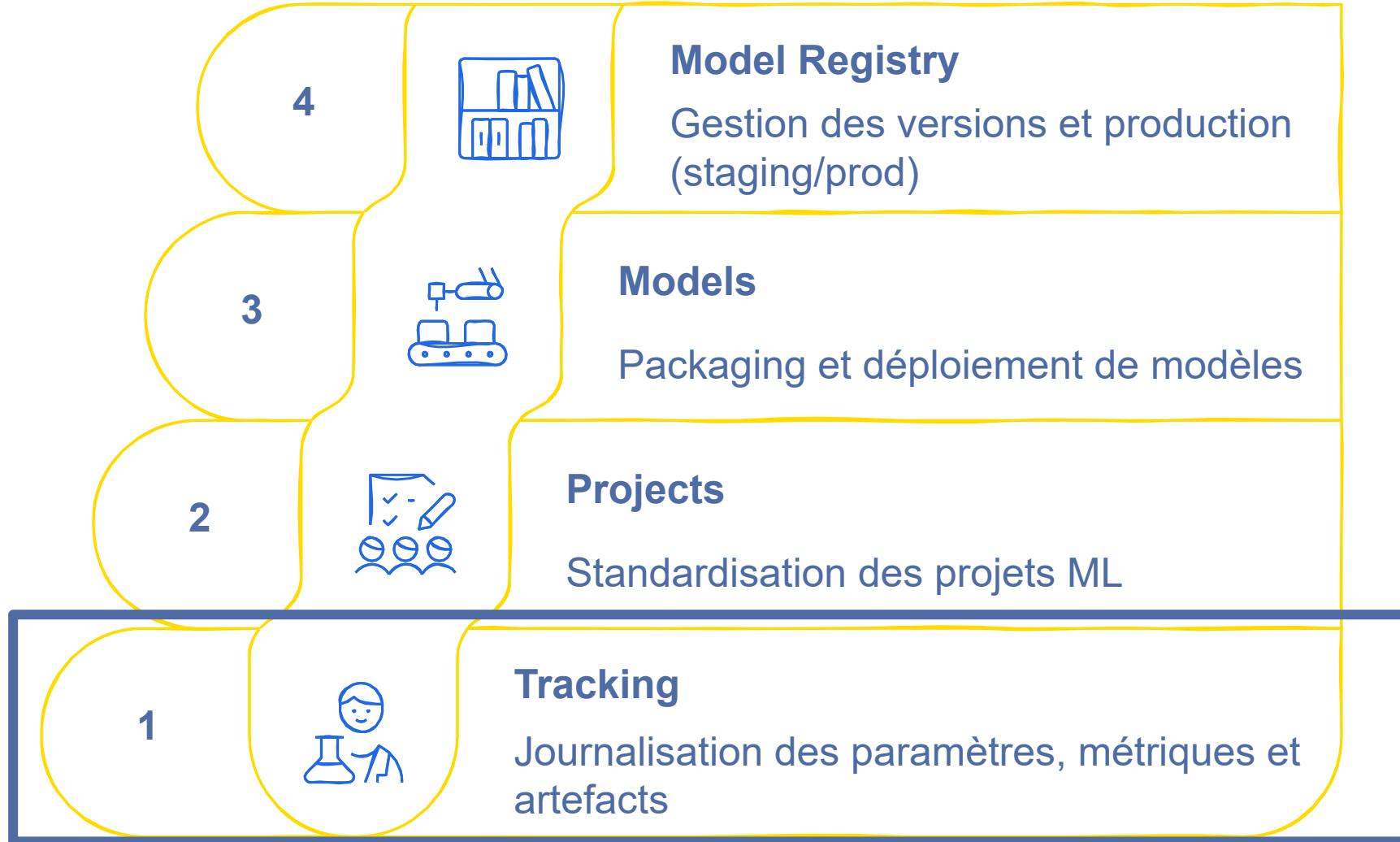
# Qu'est-ce que MLflow ?

- Permet de suivre, reproduire, packager et déployer les modèles ML
- Adapté à tous les frameworks (scikit-learn, PyTorch, etc.)

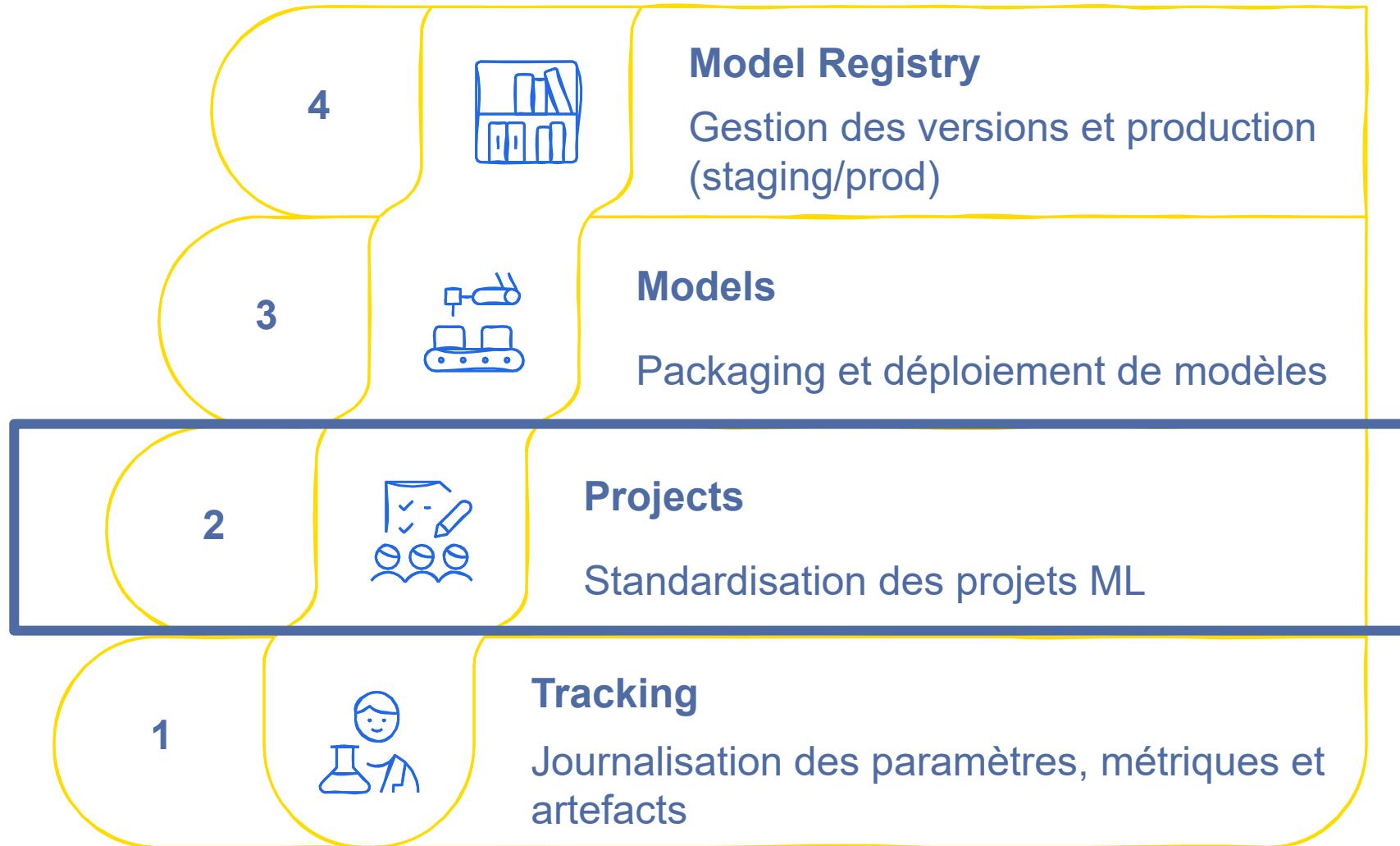
# Les 4 modules de MLflow



# Les 4 modules de MLflow



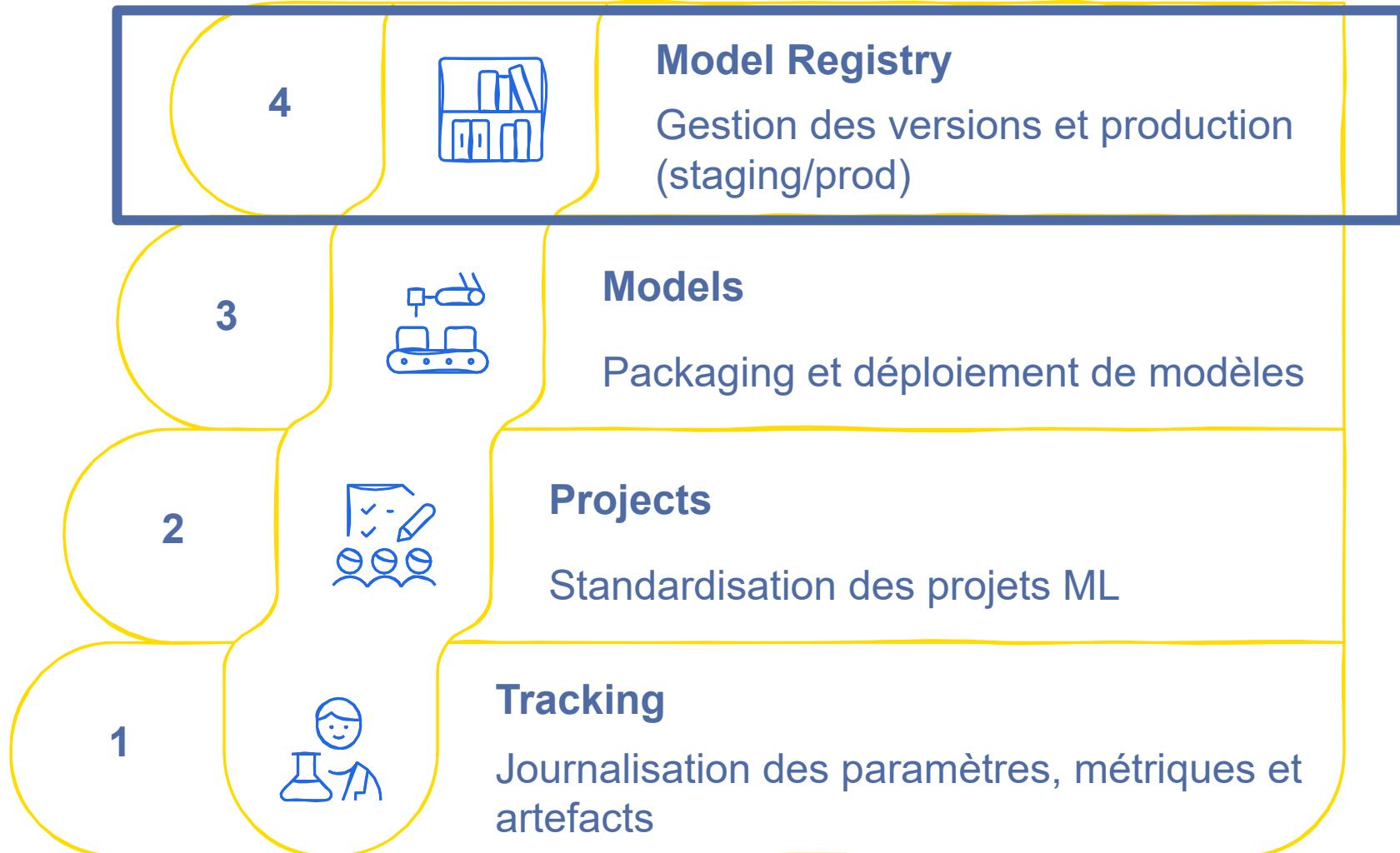
# Les 4 modules de MLflow



# Les 4 modules de MLflow



# Les 4 modules de MLflow



# Les 4 modules de MLflow



# Pourquoi MLflow en entreprise ?

# Pourquoi MLflow en entreprise ?



## Suivi des expérimentations

MLflow suit qui a fait quoi, quand, et avec quels paramètres.

# Pourquoi MLflow en entreprise ?



## Suivi des expérimentations

MLflow suit qui a fait quoi, quand, et avec quels paramètres.



## Résultats reproductibles

MLflow garantit que les résultats peuvent être reproduits de manière cohérente.

# Pourquoi MLflow en entreprise ?



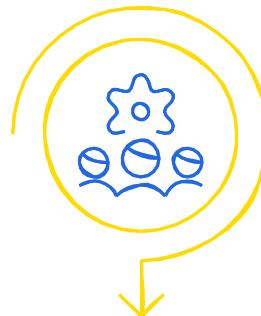
## Suivi des expérimentations

MLflow suit qui a fait quoi, quand, et avec quels paramètres.



## Résultats reproductibles

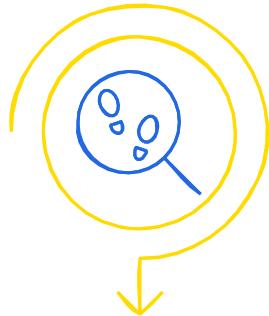
MLflow garantit que les résultats peuvent être reproduits de manière cohérente.



## Collaboration d'équipe

MLflow facilite la collaboration entre les data scientists, les ingénieurs et les parties prenantes commerciales.

# Pourquoi MLflow en entreprise ?



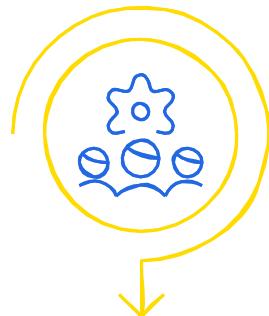
## Suivi des expérimentations

MLflow suit qui a fait quoi, quand, et avec quels paramètres.



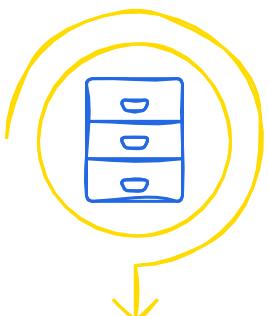
## Résultats reproductibles

MLflow garantit que les résultats peuvent être reproduits de manière cohérente.



## Collaboration d'équipe

MLflow facilite la collaboration entre les data scientists, les ingénieurs et les parties prenantes commerciales.



## Gestion des versions

MLflow améliore le contrôle des versions et les processus de mise en production.

# Cas concret : prédition du churn

# Cas concret : prédition du churn

- Comparer plusieurs modèles

# Cas concret : prédition du churn

- Comparer plusieurs modèles
- Identifier les meilleurs hyperparamètres

# Cas concret : prédition du churn

- Comparer plusieurs modèles
- Identifier les meilleurs hyperparamètres
- Suivre l'évolution des performances

# Résumé

# Résumé

- MLflow est un outil central du **MLOps**

# Résumé

- MLflow est un outil central du **MLOps**
- Il se compose de **4 modules complémentaires**

# Résumé

- MLflow est un outil central du **MLOps**
- Il se compose de **4 modules complémentaires**
- Il facilite le travail **collaboratif**, le **tracking**, et le **versionnage**

# Résumé

- MLflow est un outil central du **MLOps**
- Il se compose de **4 modules complémentaires**
- Il facilite le travail **collaboratif**, le **tracking**, et le **versionnage**
- Il sera utilisé dans ce module pour un projet **réel de prédition du churn**

# Installation & test d'intégration de MLflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Activation de l'environnement virtuel

# Plan

§ Activation de l'environnement virtuel

§ Installation de MIflow

# Plan

- § Activation de l'environnement virtuel
- § Installation de MLflow
- § Lancement de l'interface MLflow UI

# Plan

- § Activation de l'environnement virtuel
- § Installation de MLflow
- § Lancement de l'interface MLflow UI
- § Bonnes pratiques Git (.gitignore)

# Plan

- § Activation de l'environnement virtuel
- § Installation de MLflow
- § Lancement de l'interface MLflow UI
- § Bonnes pratiques Git (.gitignore)
- § Test d'intégration via  
**train\_model.py**

# Plan

- § Activation de l'environnement virtuel
- § Installation de MLflow
- § Lancement de l'interface MLflow UI
- § Bonnes pratiques Git (.gitignore)
- § Test d'intégration via  
**train\_model.py**
- § Vérification du dossier **mlruns/**

# Préparer le projet pour MLflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Ajout de la config MLflow dans config.yaml

# Plan

- § Ajout de la config MLflow dans config.yaml
- § Fonction utilitaire de logging dans utils.py

# Plan

- § Ajout de la config MLflow dans config.yaml
- § Fonction utilitaire de logging dans utils.py
- § Instrumentation de train\_model.py avec mlflow.start()

# Plan

- § Ajout de la config MLflow dans config.yaml
- § Fonction utilitaire de logging dans utils.py
- § Instrumentation de train\_model.py avec mlflow.start\_run()
- § Lancement du tracking server via mlflow ui

# Plan

- § Ajout de la config MLflow dans config.yaml
- § Fonction utilitaire de logging dans utils.py
- § Instrumentation de train\_model.py avec mlflow.step()
- § Lancement du tracking server via mlflow ui
- § Smoke test : exécution et vérification dans l'interface MLflow

# Où en est-on dans le projet ?

# Où en est-on dans le projet ?

- Ce module montre comment intégrer MLflow dans un projet MLOps réel

# Où en est-on dans le projet ?

- Ce module montre comment intégrer MLflow dans un projet MLOps réel
- Cas pratique : Prédiction du churn client.

# Où en est-on dans le projet ?

- Ce module montre comment intégrer MLflow dans un projet MLOps réel
- Cas pratique : Prédiction du churn client.
- Étape actuelle :  Préparer le projet pour permettre le suivi automatisé des entraînements

# Suivre des expérimentations avec MLflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

§ Logguer proprement les hyperparamètres et plusieurs métriques

# Plan

- § Logguer proprement les hyperparamètres et plusieurs métriques
- § Logguer des artefacts visuels

# Logguer les hyperparamètres et plusieurs métriques avec Mlflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Séparer l'entraînement des modèles baseline et XGBoost



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Logguer un modèle dans MLflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Comparer plusieurs runs et naviguer dans l'interface Mlflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Versionner, promouvoir et gérer un modèle avec Mlflow Model Registry



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Rendre l'encodage réutilisable pour l'inférence avec Mlflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Préparer le script de prédiction



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Recharger un modèle enregistré avec MLflow pour effectuer une prédiction



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Logguer les artifacts de preprocessing dans Mlflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Réutiliser les artifacts de préprocessing depuis Mlflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Organiser et nettoyer ses Runs dans l'UI MLflow



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

§ Supprimer les runs inutiles

# Plan

- § Supprimer les runs inutiles
- § Organiser les expériences intelligemment

# Plan

- § Supprimer les runs inutiles
- § Organiser les expériences intelligemment
- § Explorer le site officiel MLflow

# Explorer le rôle de DVC dans la gestion des données en MLOps



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

## § Pourquoi versionner les données

# Plan

- § Pourquoi versionner les données
- § Présentation de DVC

# Plan

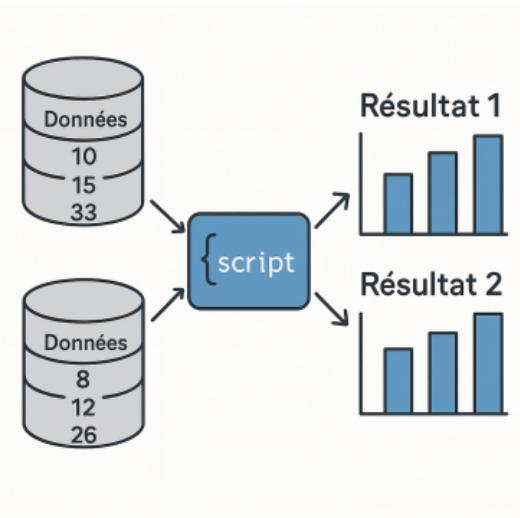
- § Pourquoi versionner les données
- § Présentation de DVC
- § Différence entre DVC et Git

# Plan

- § Pourquoi versionner les données
- § Présentation de DVC
- § Différence entre DVC et Git
- § Comment DVC complète MLflow

# Pourquoi versionner les données?

Reproduire un modèle sans les bonnes données est impossible

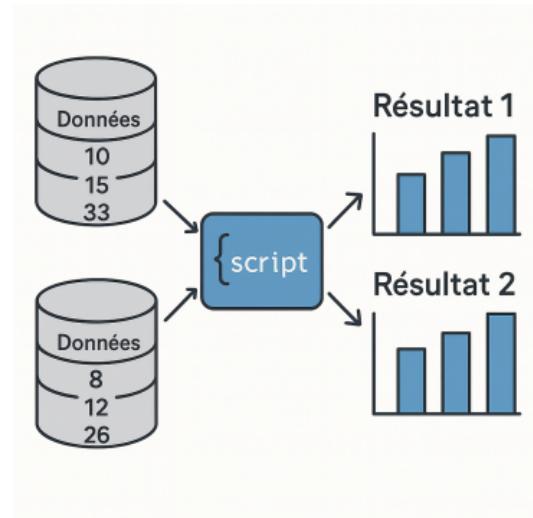


# Pourquoi versionner les données?

Reproduire un modèle sans les bonnes données est impossible

Il faut pouvoir :

- Revenir à la version exacte utilisée pour un modèle

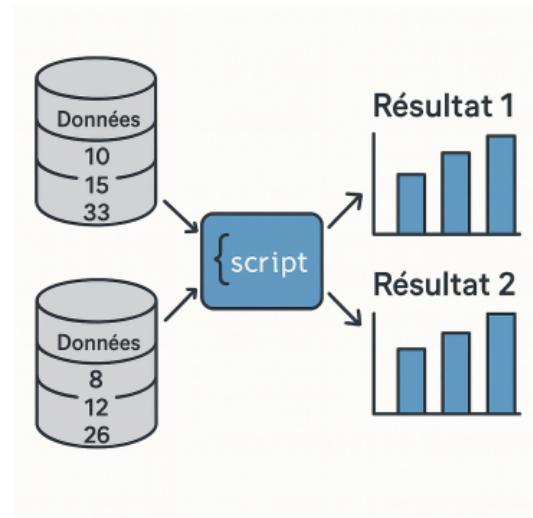


# Pourquoi versionner les données?

Reproduire un modèle sans les bonnes données est impossible

Il faut pouvoir:

- Revenir à la version exacte utilisée pour un modèle
- Comparer plusieurs jeux de données

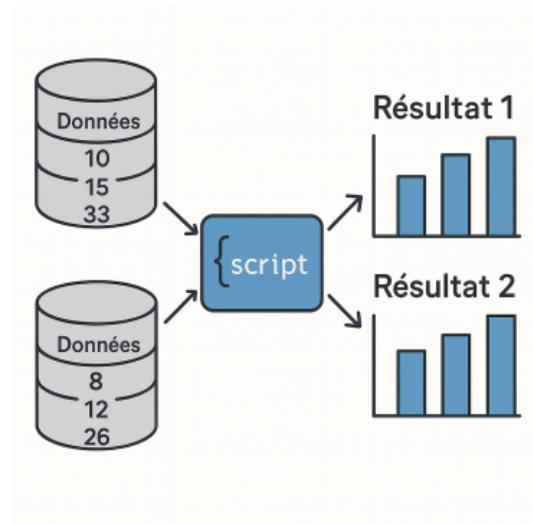


# Pourquoi versionner les données?

Reproduire un modèle sans les bonnes données est impossible

Il faut pouvoir:

- Revenir à la version exacte utilisée pour un modèle
- Comparer plusieurs jeux de données
- Travailler en équipes sans conflit



DVC c'est quoi?

# DVC c'est quoi?

**DVC = Data Version Control**

Un outil open-source conçu pour le **versionnement des données** dans les projets de ML

# DVC c'est quoi?

**DVC = Data Version Control**

- Fonctionne avec Git, mais pour les fichiers lourds

# DVC c'est quoi?

**DVC = Data Version Control**

- Fonctionne avec Git, mais pour les fichiers lourds
- Crée des fichiers de métadonnées `.dvc` pour suivre les datasets

# DVC c'est quoi?

**DVC = Data Version Control**

- Fonctionne avec Git, mais pour les fichiers lourds
- Crée des fichiers de métadonnées **.dvc** pour suivre les datasets
- Utilise un **espace de stockage externe** (local, cloud, S3...)

# DVC c'est quoi?

**DVC = Data Version Control**

- Fonctionne avec Git, mais pour les fichiers lourds
- Crée des fichiers de métadonnées **.dvc** pour suivre les datasets
- Utilise un **espace de stockage externe** (local, cloud, S3...)
- Permet de reconstruire un pipeline complet (data → modèle)

# DVC c'est quoi?

**DVC = Data Version Control**

- Fonctionne avec Git, mais pour les fichiers lourds
- Crée des fichiers de métadonnées **.dvc** pour suivre les datasets
- Utilise un **espace de stockage externe** (local, cloud, S3...)
- Permet de reconstruire un pipeline complet (data → modèle)
- Compatible avec les workflows Git : **add, commit, push, pull**

# DVC c'est quoi?

**DVC = Data Version Control**

- Fonctionne avec Git, mais pour les fichiers lourds
- Crée des fichiers de métadonnées **.dvc** pour suivre les datasets
- Utilise un **espace de stockage externe** (local, cloud, S3...)
- Permet de reconstruire un pipeline complet (data → modèle)
- Compatible avec les workflows Git : **add, commit, push, pull**

*En résumé: DVC, c'est le Git des données et des pipelines ML*

# Git vs DVC

# Git vs DVC

| Fonction                      | Git                      | Git + DVC                |
|-------------------------------|--------------------------|--------------------------|
| Suivi code                    | <input type="checkbox"/> | <input type="checkbox"/> |
| Suivi données volumineuses    | <input type="checkbox"/> | <input type="checkbox"/> |
| Lien entre données et modèles | <input type="checkbox"/> | <input type="checkbox"/> |

# DVC et MLflow : complémentaire

# DVC et MLflow : complémentaire

Deux outils, deux missions :

# DVC et MLflow : complémentaire

Deux outils, deux missions :

- MLflow : suivi des **expériences**, hyperparamètres, métriques, artefacts

# DVC et MLflow : complémentaire

Deux outils, deux missions :

- **MLflow** : suivi des **expériences**, hyperparamètres, métriques, artefacts
- **DVC** : suivi **versionné** des données, modèles, et **pipelines reproductibles**

# DVC et MLflow : complémentaire

Ensemble, ils permettent de :

- Revenir à un état exact (code + données + modèle)
- Comprendre et reproduire un résultat précis
- Collaborer efficacement sur tout le cycle de vie ML

# DVC et MLflow : complémentaire

Ensemble, ils permettent de :

- Revenir à un état exact (code + données + modèle)
- Comprendre et reproduire un résultat précis
- Collaborer efficacement sur tout le cycle de vie ML

# DVC et MLflow : complémentaire

Ensemble, ils permettent de :

- Revenir à un état exact (code + données + modèle)
- Comprendre et reproduire un résultat précis
- Collaborer efficacement sur tout le cycle de vie ML

# Résumé

# Résumé

- Les données ne peuvent pas être versionnées efficacement avec Git seul

# Résumé

- Les données ne peuvent pas être versionnées efficacement avec Git seul
- DVC apporte : reproductibilité, traçabilité, collaboration

# Résumé

- Les données ne peuvent pas être versionnées efficacement avec Git seul
- DVC apporte : reproductibilité, traçabilité, collaboration
- DVC complète parfaitement MLflow pour structurer un pipeline MLOps robuste

# Initialiser DVC pour suivre les données du projet



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Plan

# Plan

§ Installer DVC

# Plan

§ Installer DVC

§ Initialiser DVC dans le projet

# Plan

§ Installer DVC

§ Initialiser DVC dans le projet

§ Intégrer DVC à git

# Plan

§ Installer DVC

§ Initialiser DVC dans le projet

§ Intégrer DVC à git

§ Suivre et versionner les données

# Créer un remote DVC



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Transférer les données dans le remote DVC



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Simuler un collaborateur avec Git et DVC



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Créer un pipeline DVC en construisant le stage de prétraitement



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Créer un pipeline DVC en construisant le stage d'entraînement



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

# Créer un pipeline DVC et exécuter l'ensemble des étapes



Arthur Cartel FOAHOM GOUABOU  
Data Scientist & Chercheur en IA | Formateur IT

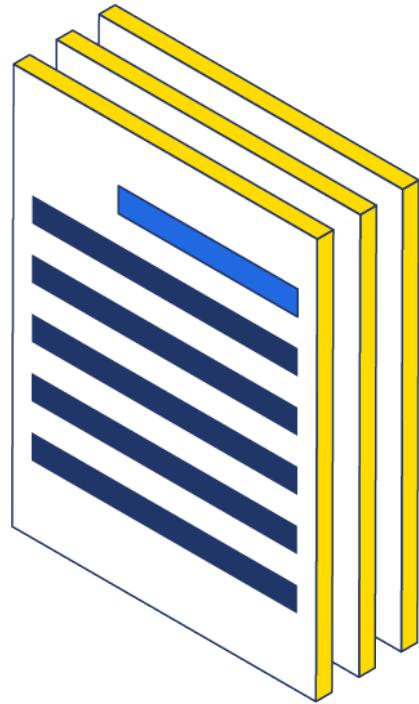
# Conclusion



**Arthur Cartel Foahom Gouabou**

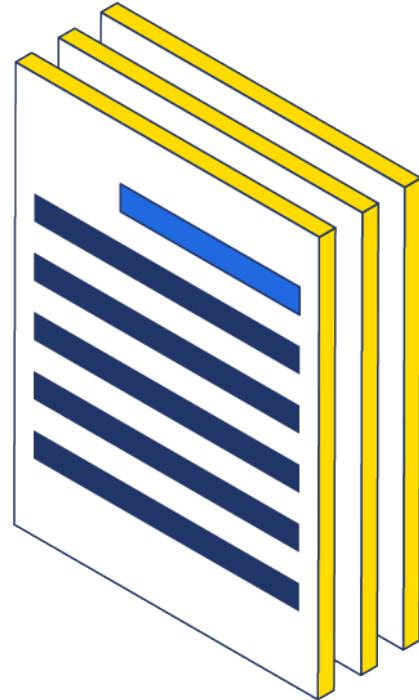
Data Scientist & Chercheur en IA | Formateur IT

# Récapitulatif des étapes clés



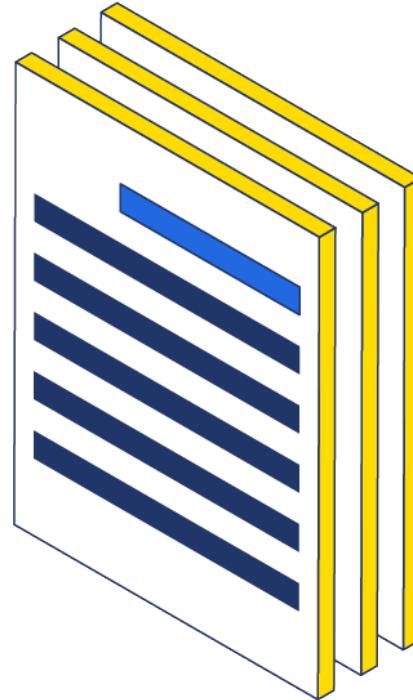
# Récapitulatif des étapes clés

- Compréhension des concepts du MLOps et cycle de vie ML



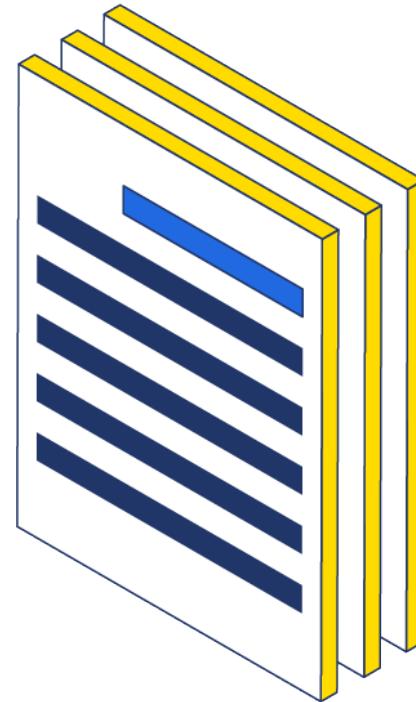
# Récapitulatif des étapes clés

- Compréhension des concepts du MLOps et cycle de vie ML
- Installation et configuration d'un environnement complet



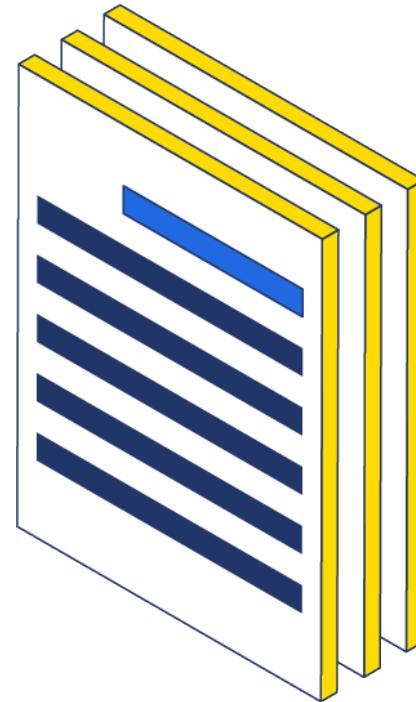
# Récapitulatif des étapes clés

- Compréhension des concepts du MLOps et cycle de vie ML
- Installation et configuration d'un environnement complet
- Structuration d'un pipeline ML reproductible



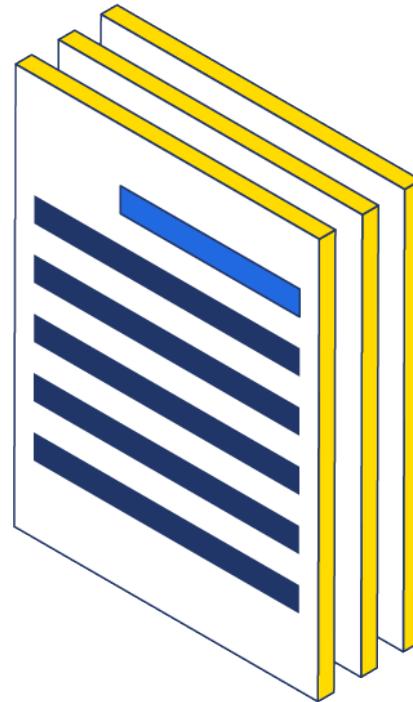
# Récapitulatif des étapes clés

- Compréhension des concepts du MLOps et cycle de vie ML
- Installation et configuration d'un environnement complet
- Structuration d'un pipeline ML reproductible
- Suivi des expériences avec Mlflow



# Récapitulatif des étapes clés

- Compréhension des concepts du MLOps et cycle de vie ML
- Installation et configuration d'un environnement complet
- Structuration d'un pipeline ML reproductible
- Suivi des expériences avec MLflow
- Gestion et versioning des données avec DVC



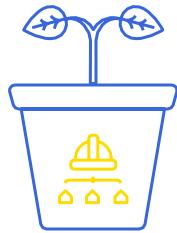
# Compétences acquises

# Compétences acquises



Établir les bases

# Compétences acquises

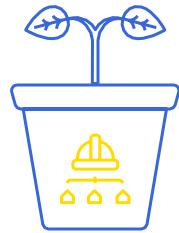


Établir les bases



Mettre en place un  
environnement  
reproductible

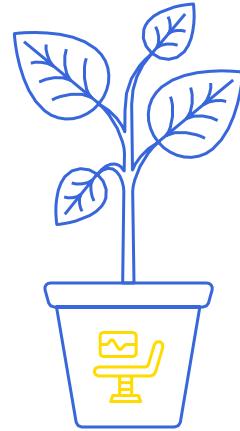
# Compétences acquises



Établir les bases

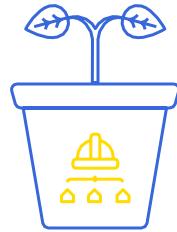


Mettre en place un  
environnement  
reproductible



Suivre et gérer  
les expériences

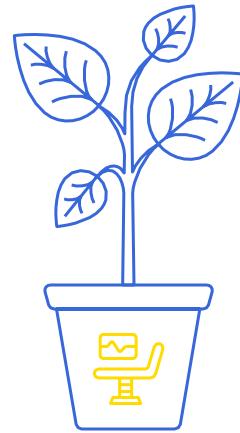
# Compétences acquises



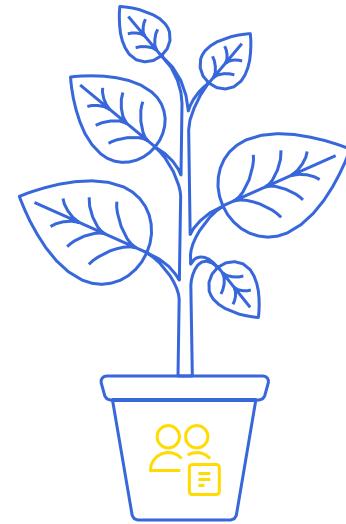
Établir les bases



Mettre en place un  
environnement  
reproductible



Suivre et gérer  
les expériences



Versionner et  
collaborer sur les  
données

# Prochaine formation

# Prochaine formation

---



# Prochaine formation

---



# Prochaine formation

---



