



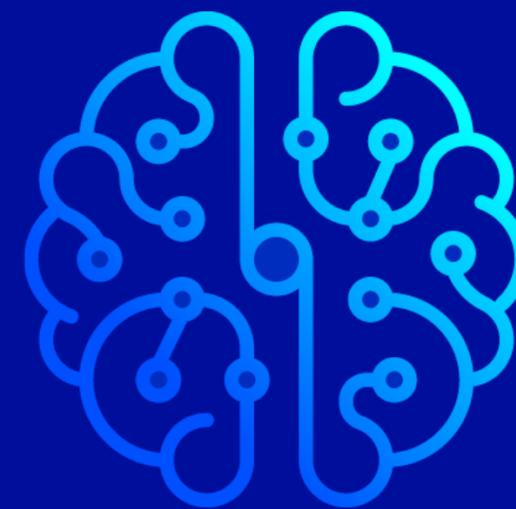
Viens dompter ta première IA en Python





MERCI !!!

Qu'allons-nous voir aujourd'hui ?



01 Les principes dans l'IA

02 Kit de survie Python

03 CDE

04 Les ressources utilisées chez OVHcloud

05 Développer à l'aide d'un notebook, d'un job et d'une app

Qui sommes-nous ?



Stéphane Philippart

Developer Advocate

OVHcloud



Mathieu BUSQUET

Machine Learning Engineer

OVHcloud

Stéphane Philippart



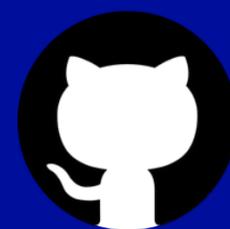
📌 🥑 DeveloperAdvocate@OVHCloud 🦄
📌 Co-créateur de [TADx](#) (meetup à Tours) 

🧠 Padawan Intelligence Artificielle 🏕️

```
whoami
stef
```



[@wildagsx](#)



[philippart-s](#)



[philippart-s.github.io/blog](#)



[philippartstephane](#)

Mathieu Busquet



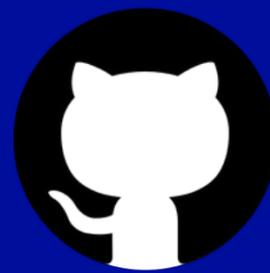
Machine Learning Engineer @OVHCloud



AI Solutions Team



[@mat_bsqt](#)



[MathieuBsqt](#)



[Mathieu Busquet](#)

Qui êtes-vous ?



Qui êtes-vous ?

- Dev



Qui êtes-vous ?

- Dev
- Dev Python



Qui êtes-vous ?

- Dev
- Dev Python
- Data Scientist



Qui êtes-vous ?

- Dev
- Dev Python
- Data Scientist
- Machine Learning Engineer



Qui êtes-vous ?

- Dev
- Dev Python
- Data Scientist
- Machine Learning Engineer
- Autre



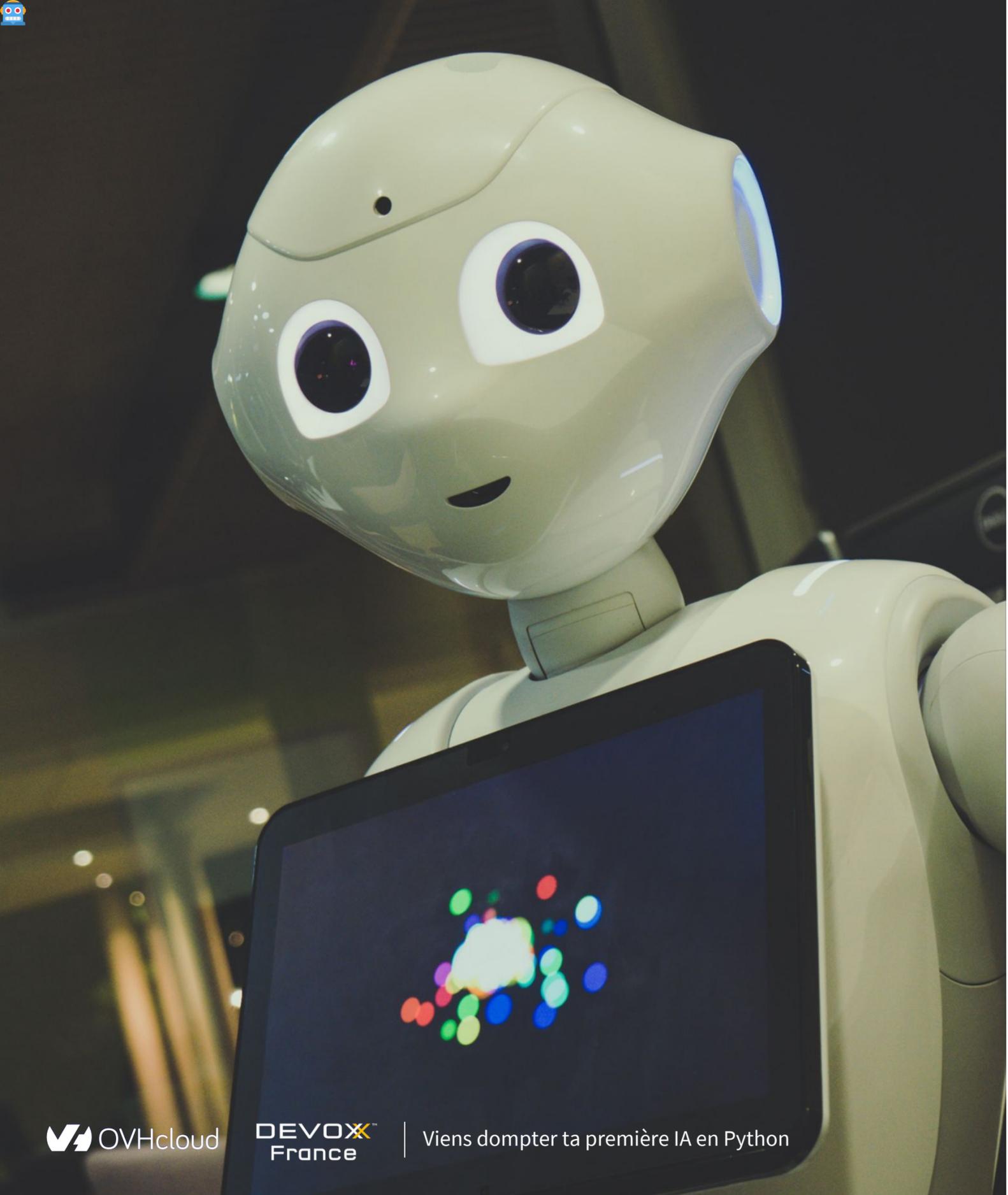


Pense-bête et liens



<https://bit.ly/101-ai-lab>





Intelligence Artificielle



L'IA dans notre quotidien

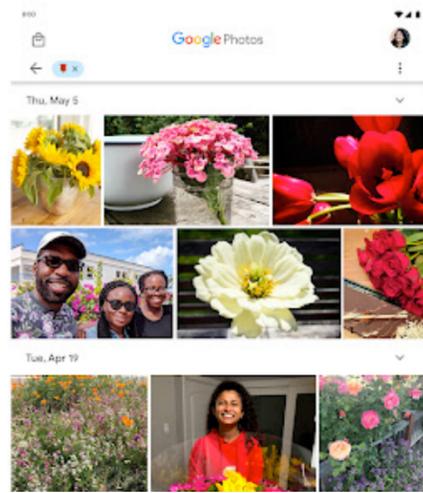
NETFLIX

Recommandations personnalisées de contenu



Google

Taggage automatique, recherche de contenus



TESLA

Conduite assistée et autonome



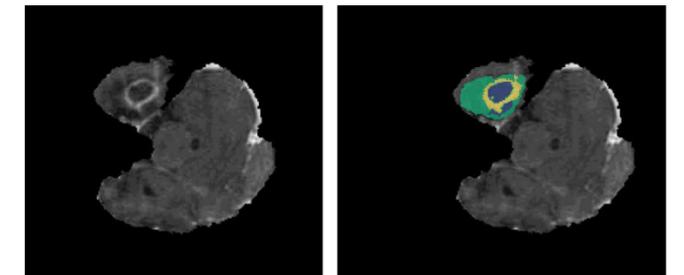
Hey Siri

Assistant vocal



zebra
MEDICAL.VISION

Aide aux diagnostics médicaux



Architecture, Assurance, Énergie, Industrie, Télécom, Transport, Vente, ...



Les champs de l'IA

Intelligence
Artificielle (IA)

👉 **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain



Les champs de l'IA

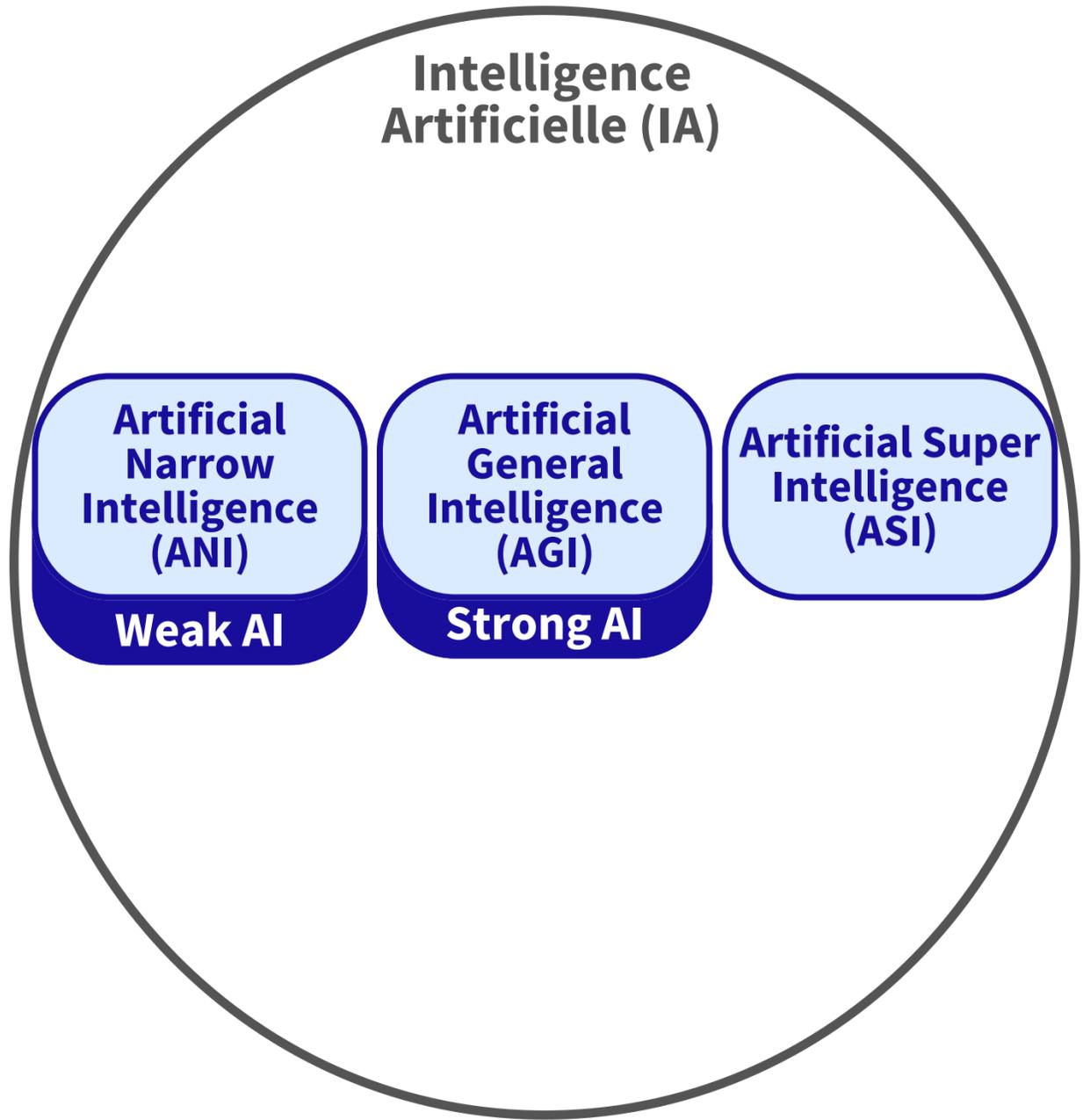
Intelligence
Artificielle (IA)

👉 **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain

- Résolution de problème
- Apprendre des expériences passées
- Perception (images, sons)
- Prise de décision



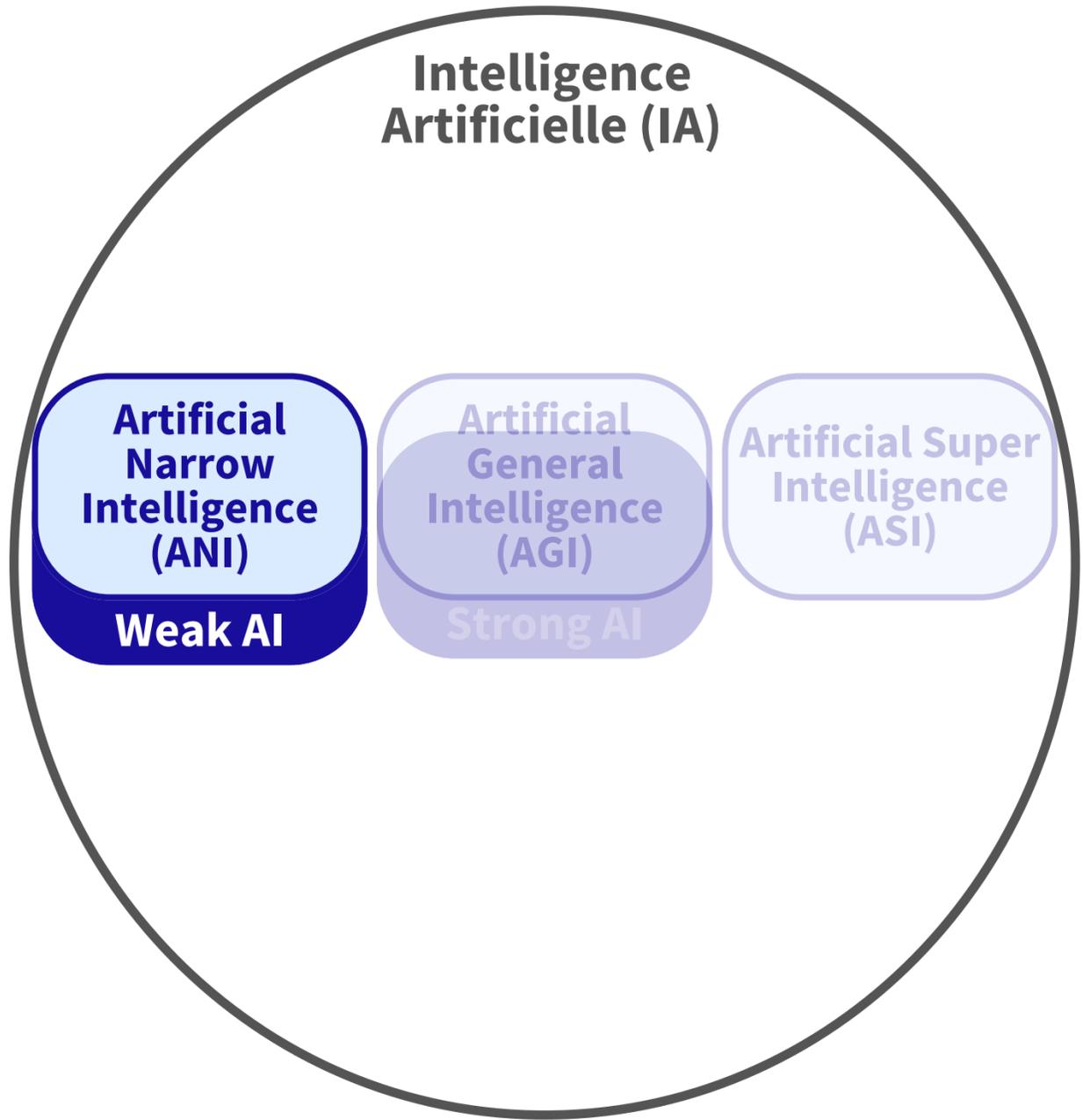
Les champs de l'IA



👉 **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain



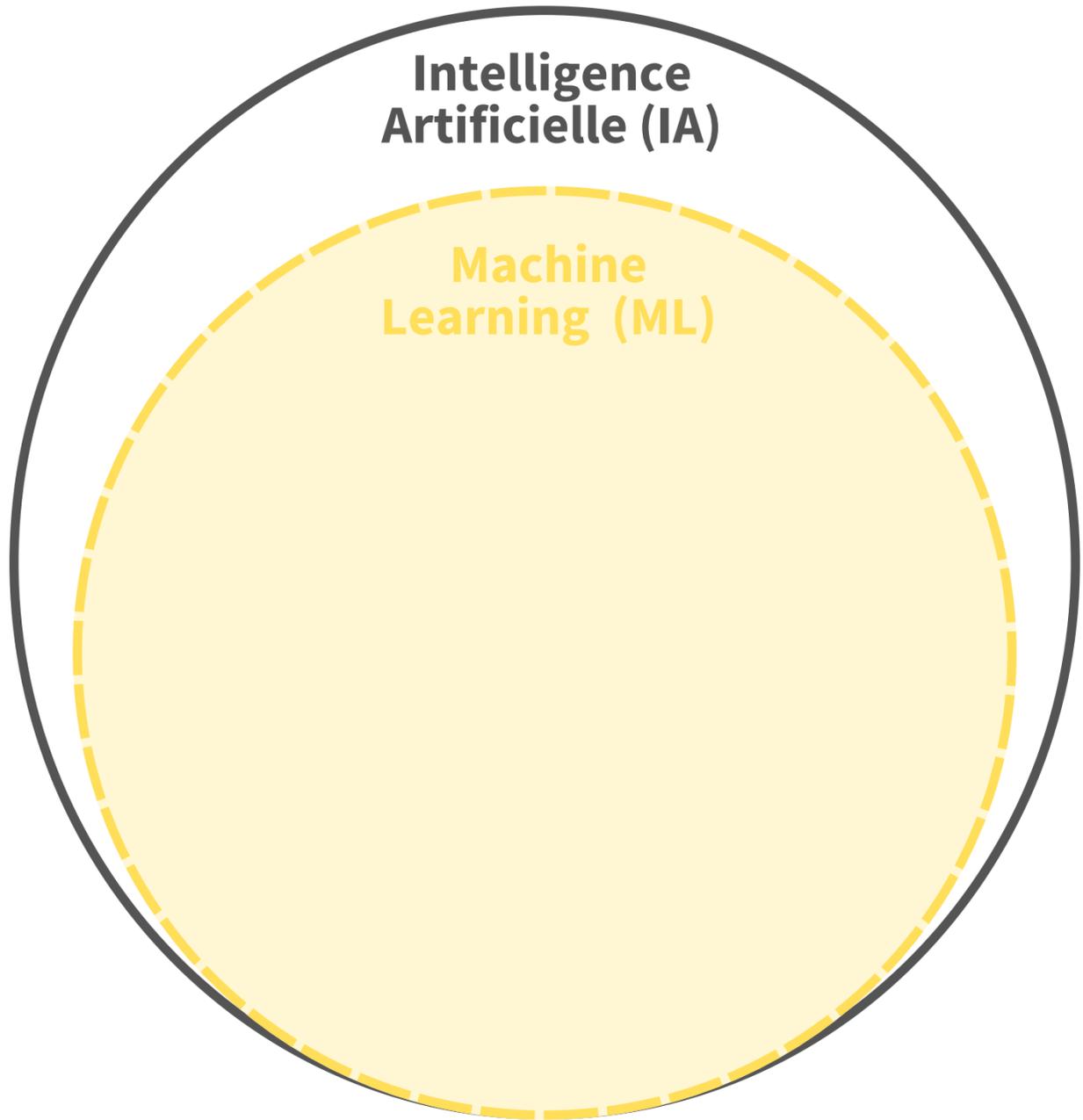
Les champs de l'IA



👉 **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain



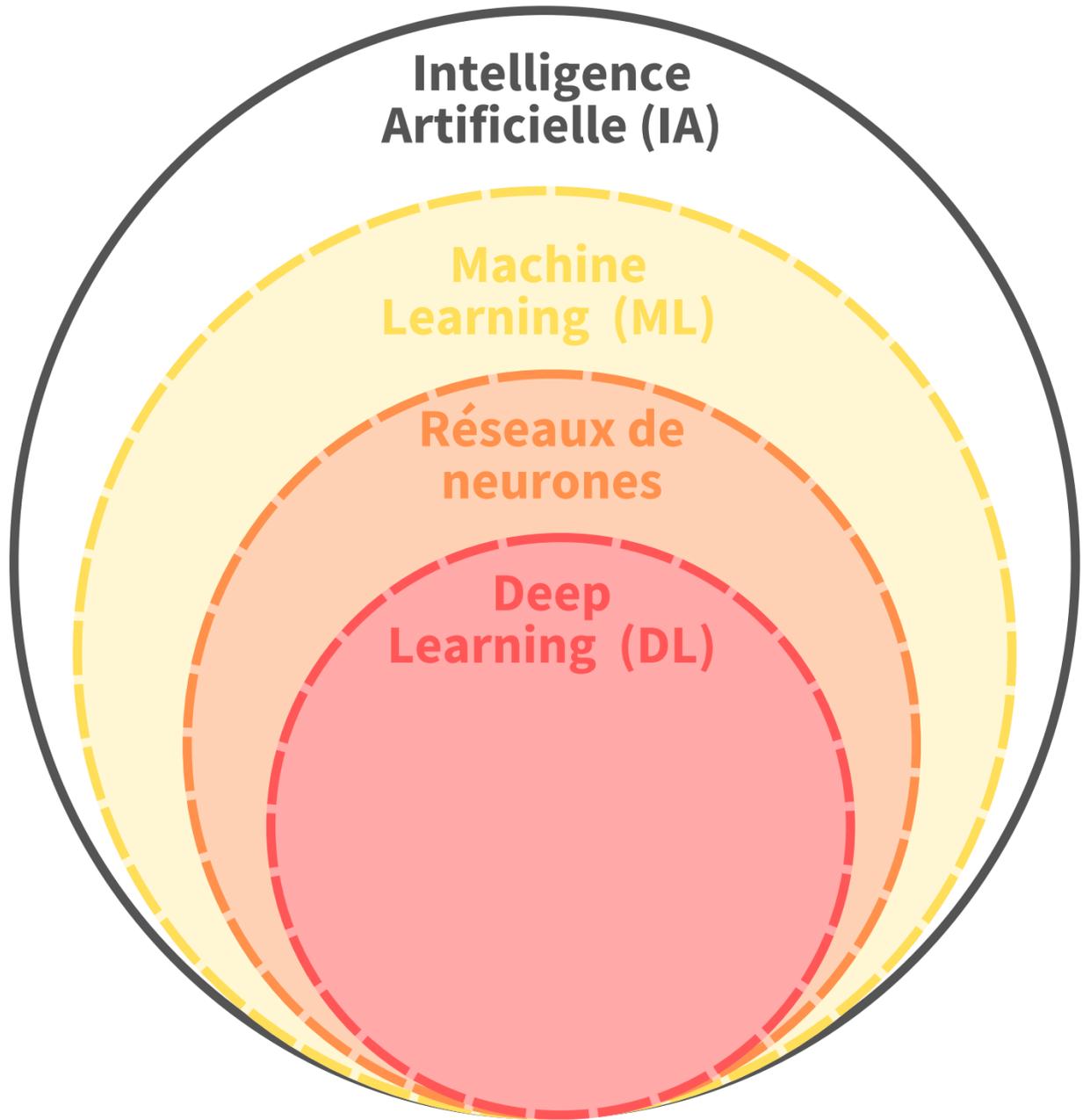
Les champs de l'IA



- 👉 **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain
- 👉 **ML** : Utilisation d’algorithmes permettant aux machines d’apprendre par l’ingestion de données, sans être explicitement programmées.



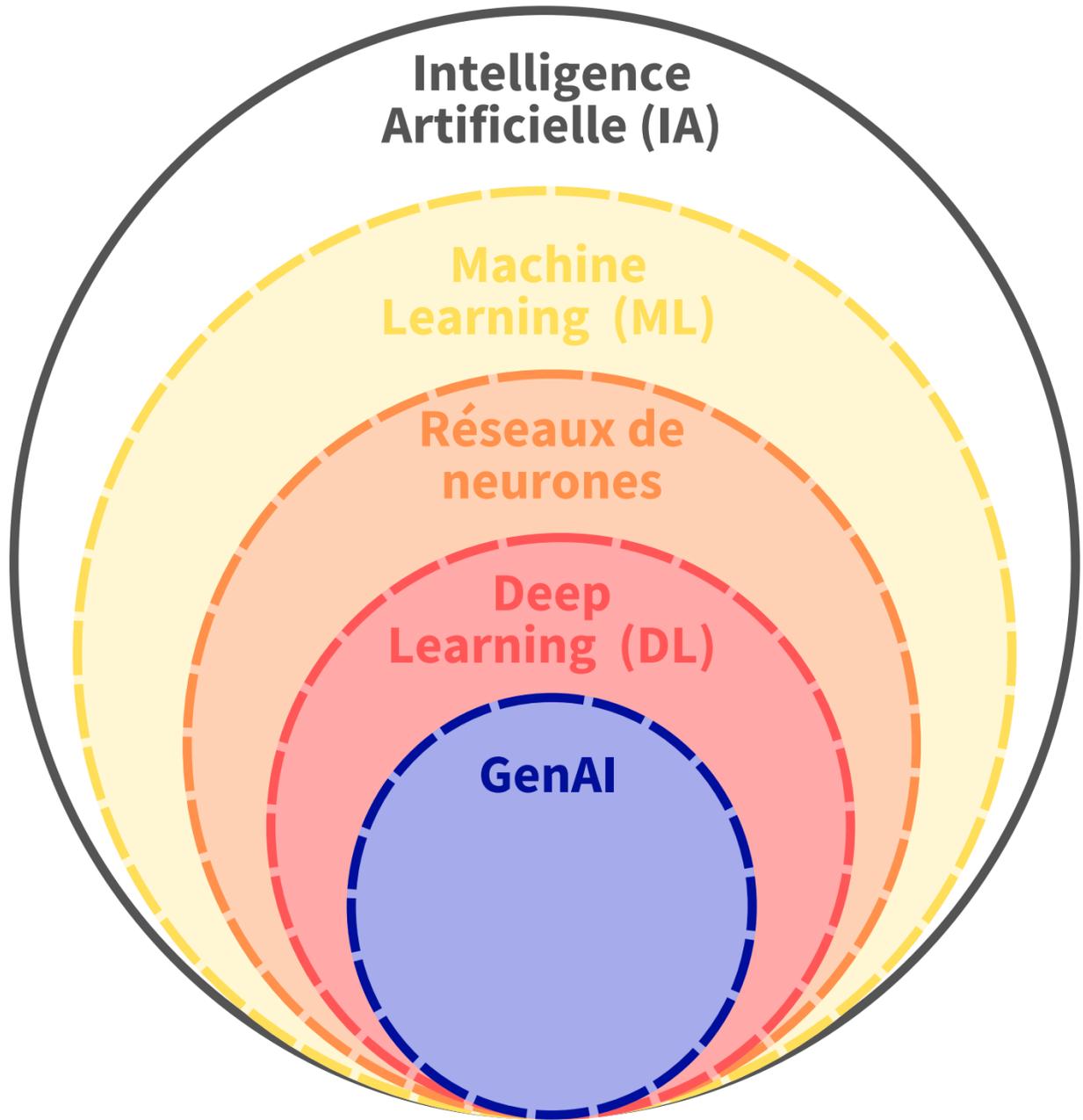
Les champs de l'IA



- 👉 **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain
- 👉 **ML** : Utilisation d’algorithmes permettant aux machines d’apprendre par l’ingestion de données, sans être explicitement programmées.
- 👉 **DL** : Utilisation de réseaux de neurones pour apprendre à partir de données non structurées, et plus complexes



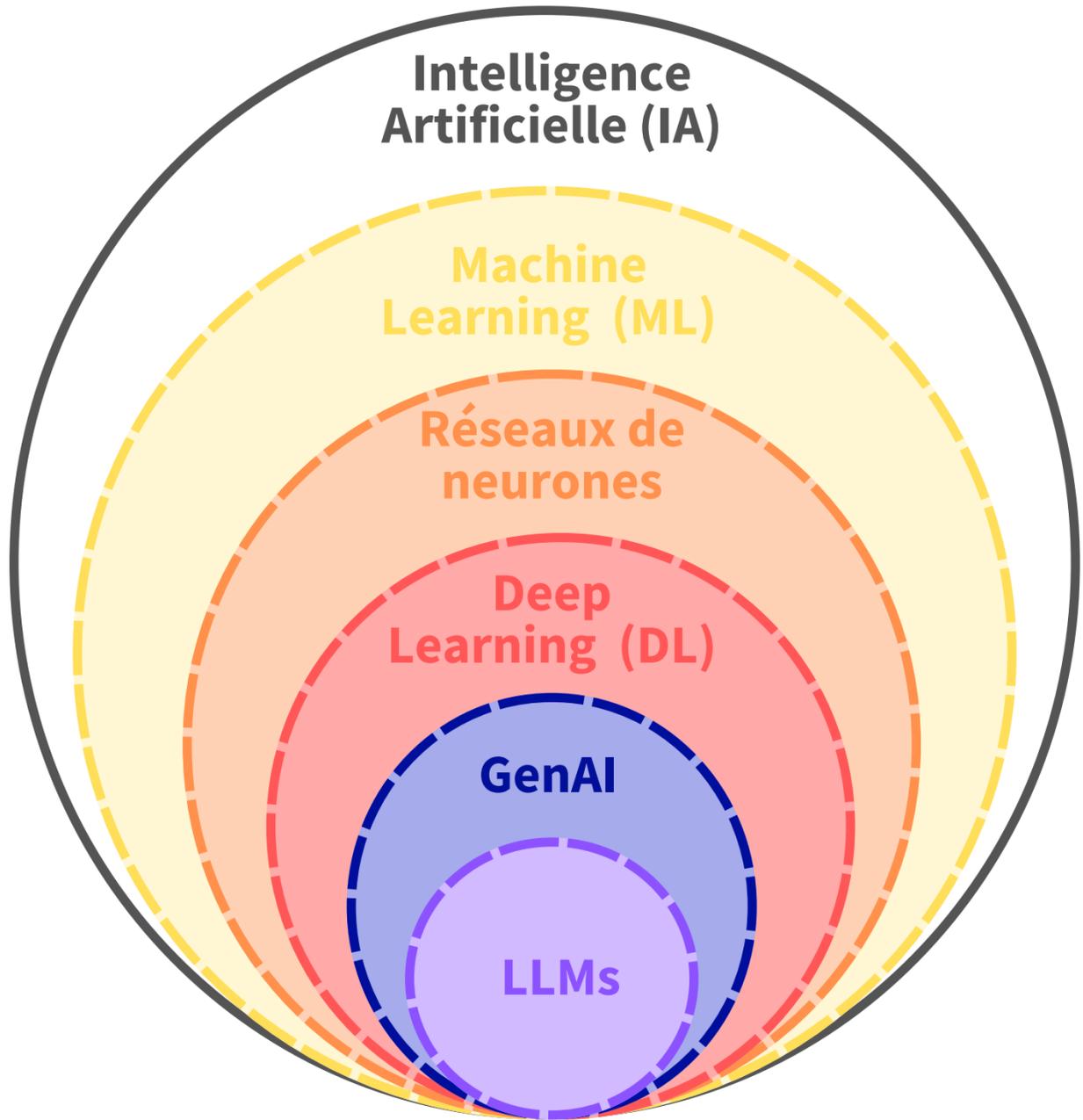
Les champs de l'IA



- ☞ **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain
- ☞ **ML** : Utilisation d’algorithmes permettant aux machines d’apprendre par l’ingestion de données, sans être explicitement programmées.
- ☞ **DL** : Utilisation de réseaux de neurones pour apprendre à partir de données non structurées, et plus complexes

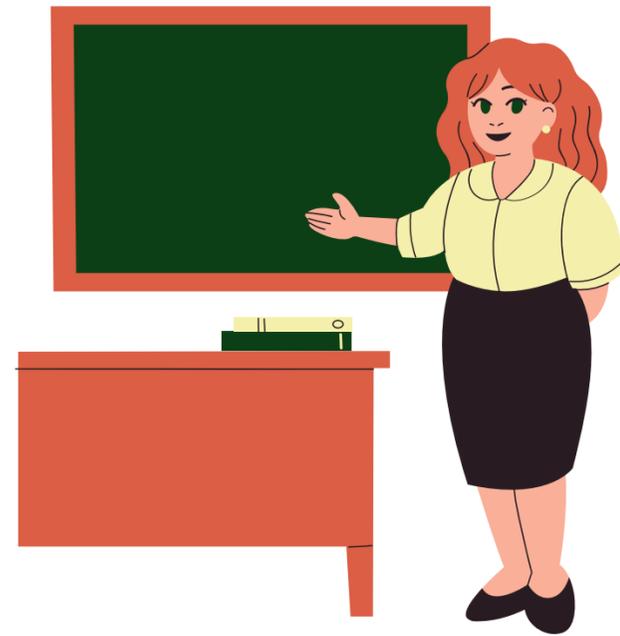


Les champs de l'IA



- ☞ **IA** : Techniques pour simuler les capacités “intelligentes” d’un être humain
- ☞ **ML** : Utilisation d’algorithmes permettant aux machines d’apprendre par l’ingestion de données, sans être explicitement programmées.
- ☞ **DL** : Utilisation de réseaux de neurones pour apprendre à partir de données non structurées, et plus complexes

Les grandes familles d'algorithmes en ML



Apprentissage supervisé

Classification et régression

La donnée est labellisée



Oiseau

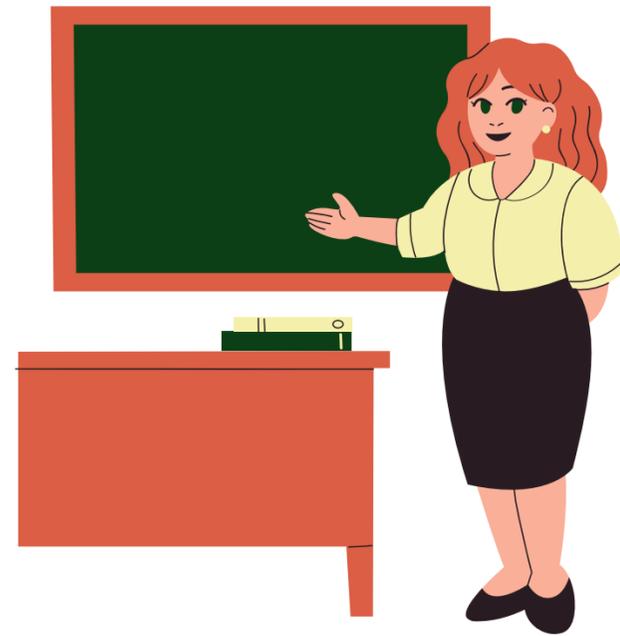


Chien



Chat

Les grandes familles d'algorithmes en ML



Apprentissage supervisé

Classification et régression

La donnée est labellisée



Oiseau



Chien



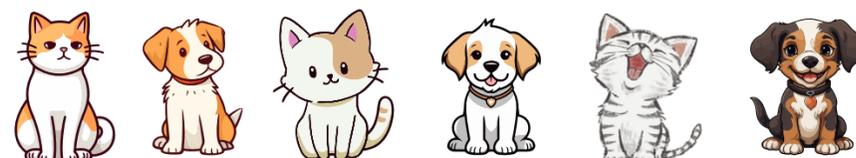
Chat



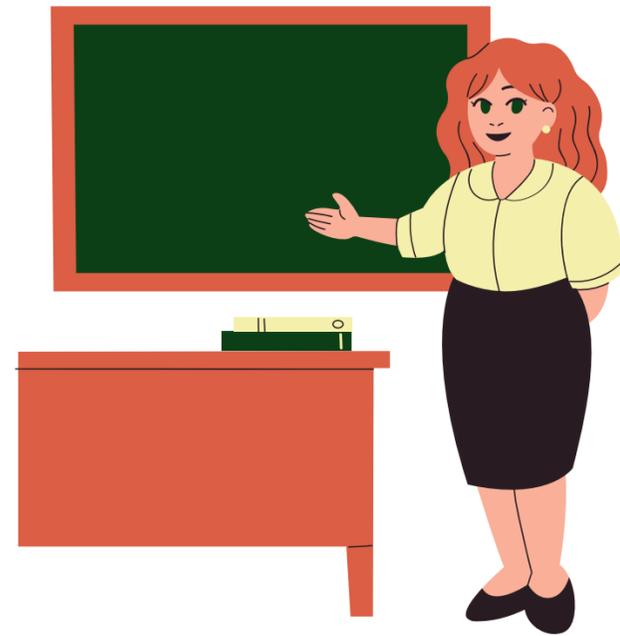
Apprentissage non supervisé

Clustering

La donnée n'est pas labellisée
C'est au modèle d'identifier les relations entre les données
Chercher la proximité entre des documents, des individus



Les grandes familles d'algorithmes en ML



Apprentissage supervisé

Classification et régression

La donnée est labellisée



Oiseau



Chien



Chat



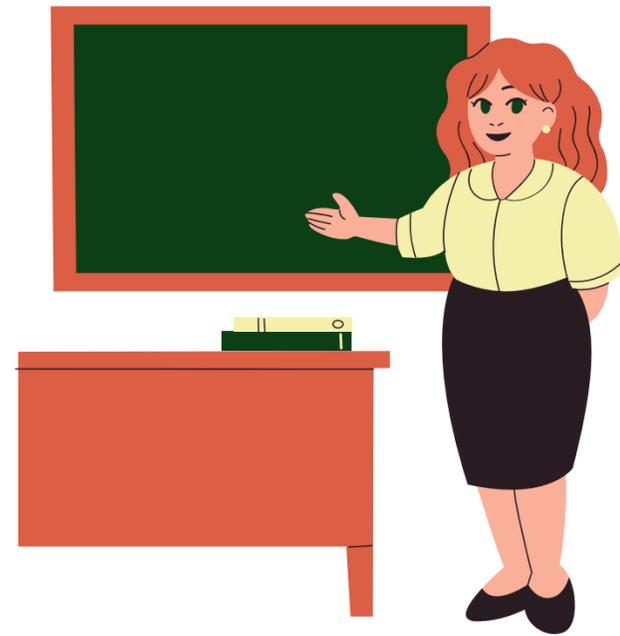
Apprentissage non supervisé

Clustering

La donnée n'est pas labellisée
C'est au modèle d'identifier les relations entre les données
Chercher la proximité entre des documents, des individus



Les grandes familles d'algorithmes en ML



Apprentissage supervisé

Classification et régression

La donnée est labellisée



Oiseau



Chien



Chat



Apprentissage non supervisé

Clustering

La donnée n'est pas labellisée
C'est au modèle d'identifier les relations entre les données
Chercher la proximité entre des documents, des individus



Apprentissage par renforcement

Les décisions du modèle sont encadrées par un système de récompenses et de pénalités

Robots (Jeux vidéo)

Sans donnée ... pas d'intelligence !



- 🖌️ La plupart du temps les données sont brutes : il faudra les nettoyer, les pré-traiter, les transformer, ...
- 📀 Il va falloir les fusionner en un Dataset (en gros une base de données pour IA)
- 🧪 Un dataset contient les données d'apprentissage, de validation et de test
- 📊 Enfin, ce sont des vecteurs et matrices qui seront manipulés par le modèle



Exemple d'un dataset



Label correspondant :



Classe personne

Coordonnées du carré
(centre_x, centre_y , largeur, hauteur)



Image labellisée



Les modèles dans l'IA



-  C'est le cerveau de votre application
-  C'est ici que l'on retrouve les formules mathématiques
-  Différents en fonction des tâches de machine learning

Deep learning & Réseau de neurones

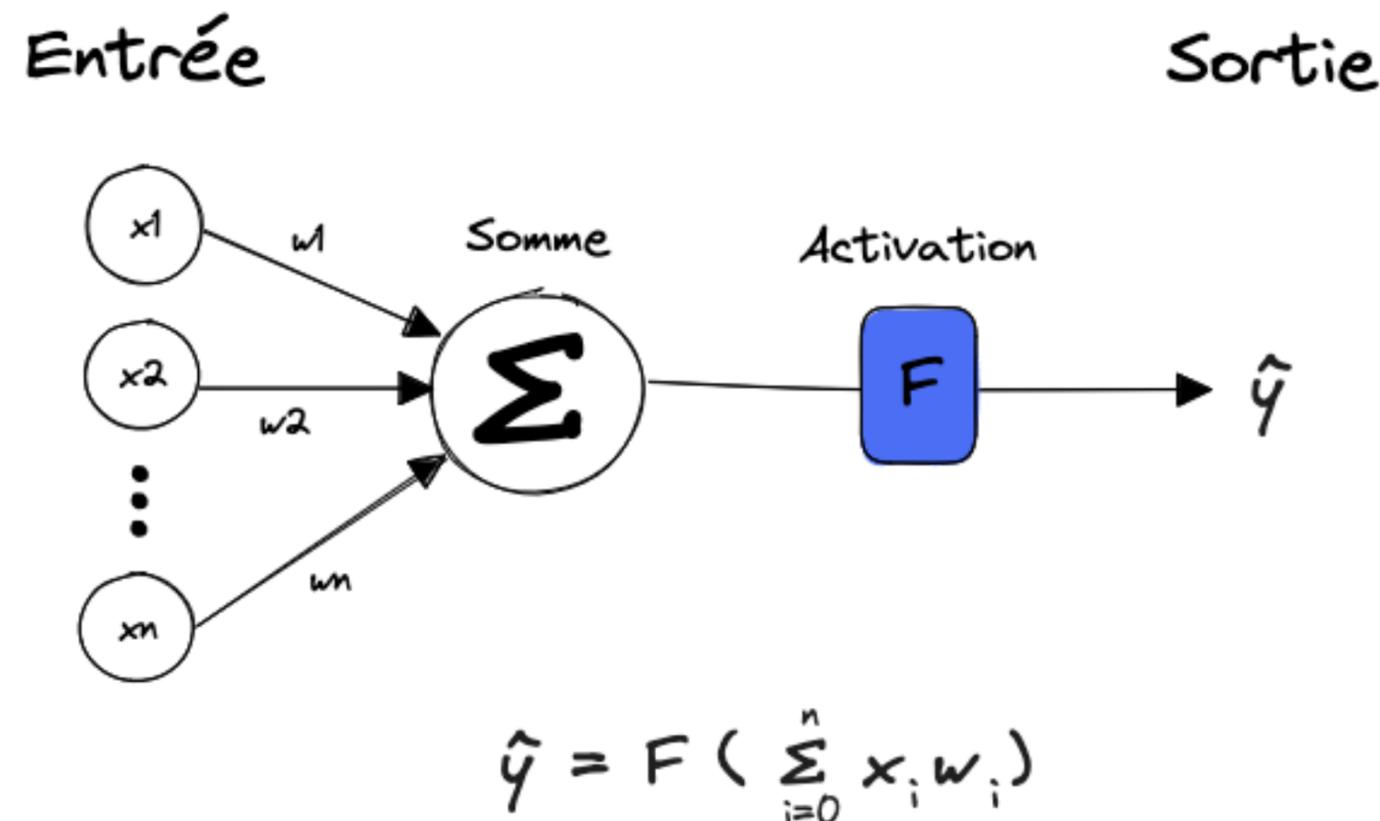


- 💪 L'étape d'après ... Plus puissante, plus complexe
- 🧠 Analogie avec les neurones humain plutôt fausse
- ↔ Essentiellement due au mécanisme d'inter-connexions

⚖️ w^* sont les poids et permettent d'ajuster le comportement du réseau

🐛 b est un biais pour rendre le modèle "plus réaliste" et influencer sur la fonction d'activation

🔄 F est une fonction d'activation permettant de modéliser des relations plus complexes



Deep learning & Réseau de neurones

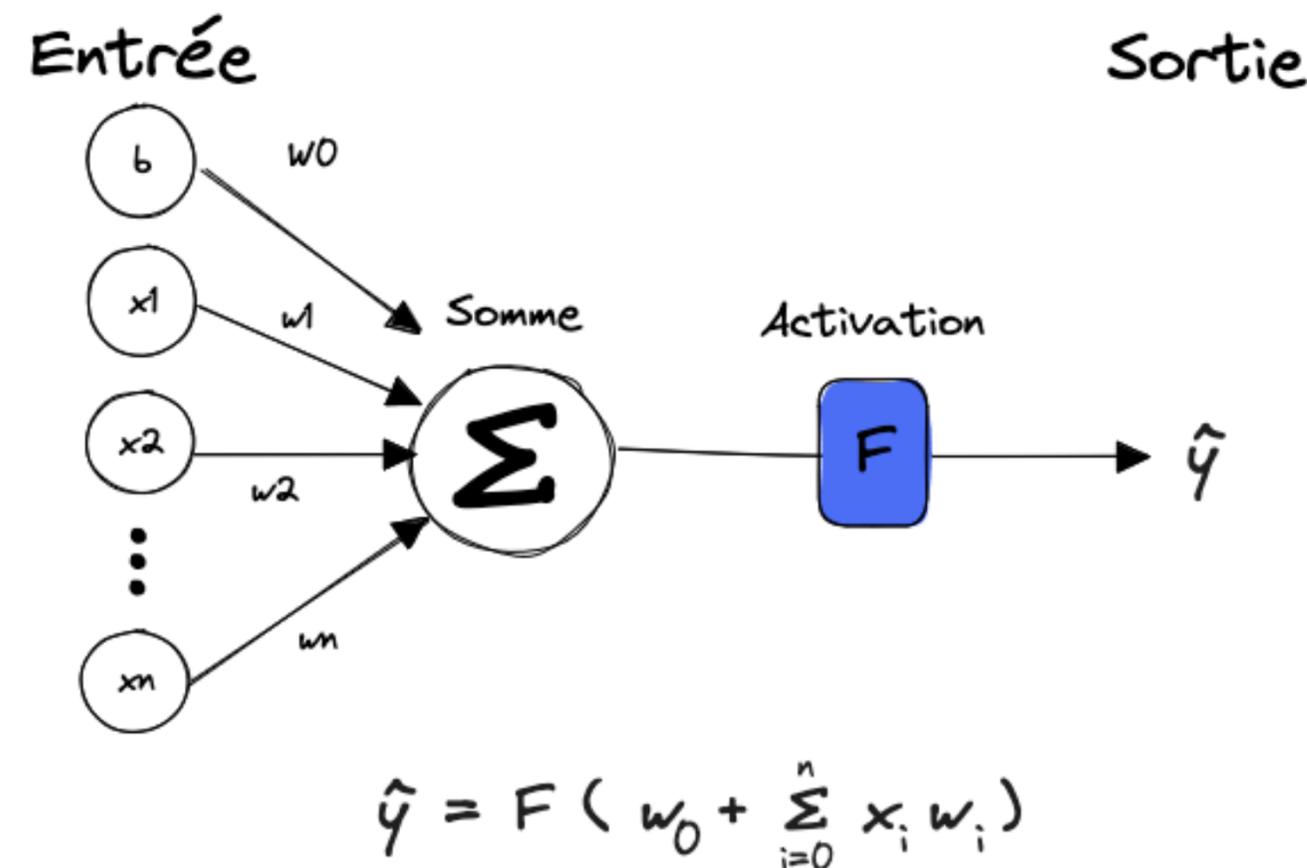


- 💪 L'étape d'après ... Plus puissante, plus complexe
- 🧠 Analogie avec les neurones humain plutôt fausse
- ↔ Essentiellement due au mécanisme d'inter-connexions

⚖️ w^* sont les poids et permettent d'ajuster le comportement du réseau

🐛 b est un biais pour rendre le modèle "plus réaliste" et influencer sur la fonction d'activation

🔄 F est une fonction d'activation permettant de modéliser des relations plus complexes

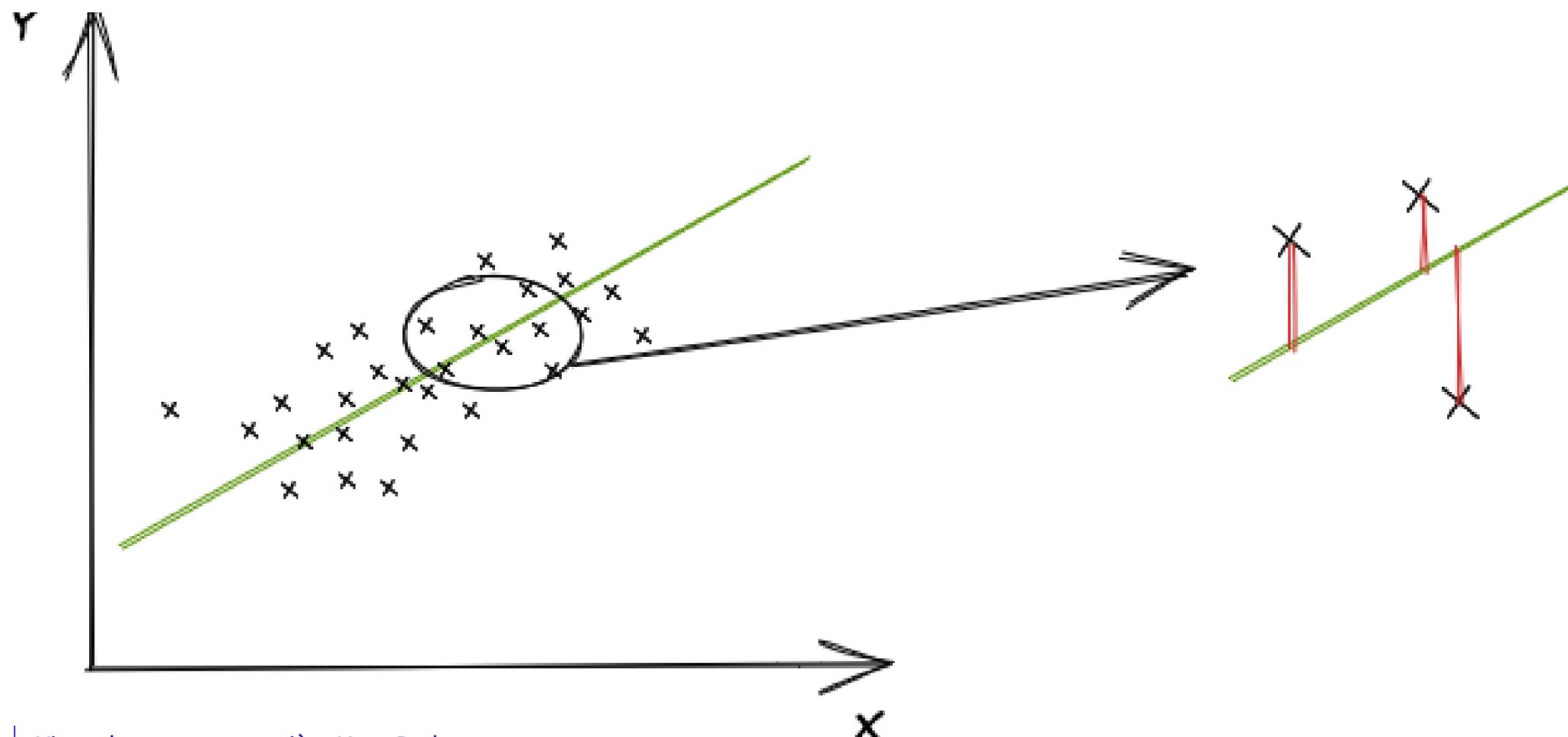




Le coût associé au modèle

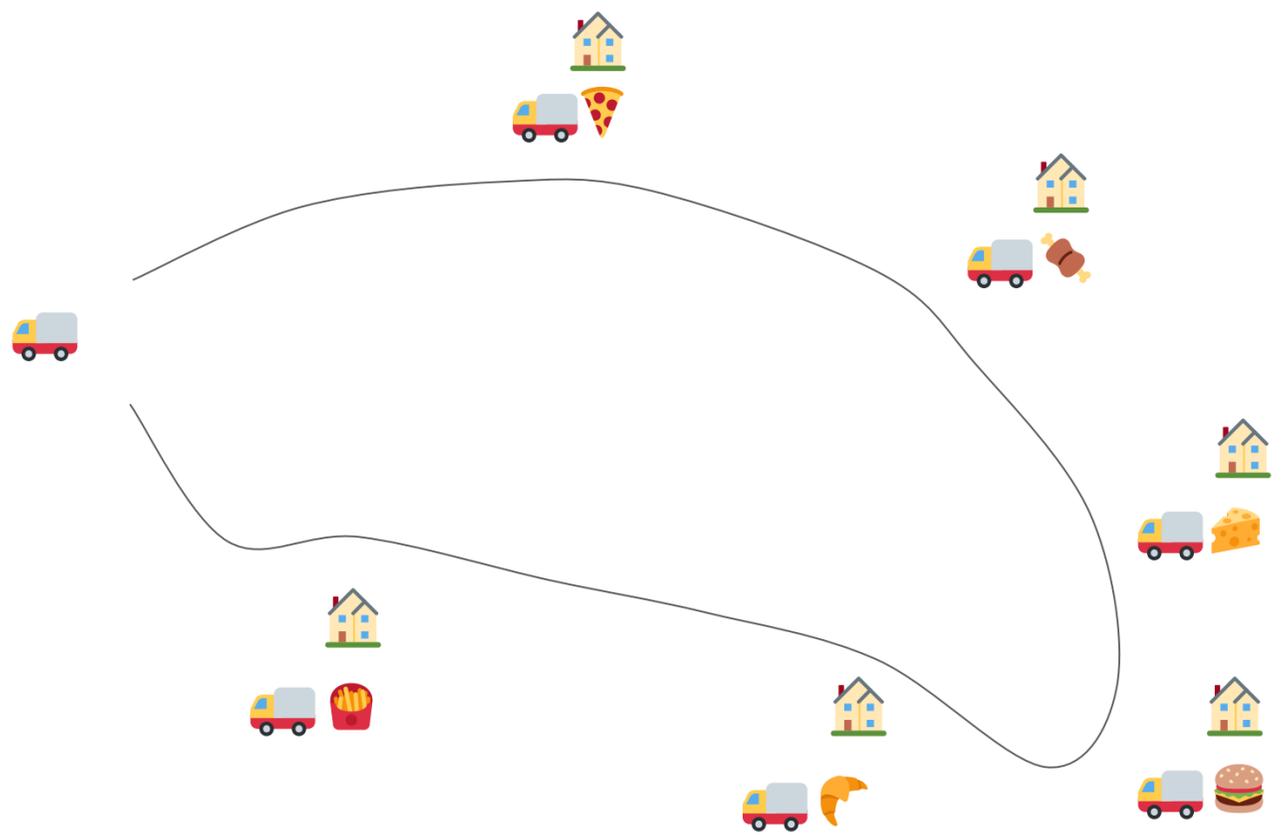


- ✨ C'est ce qui va permettre de connaître la qualité d'un modèle
- 📏 C'est l'écart entre la valeur rendue et la valeur idéale

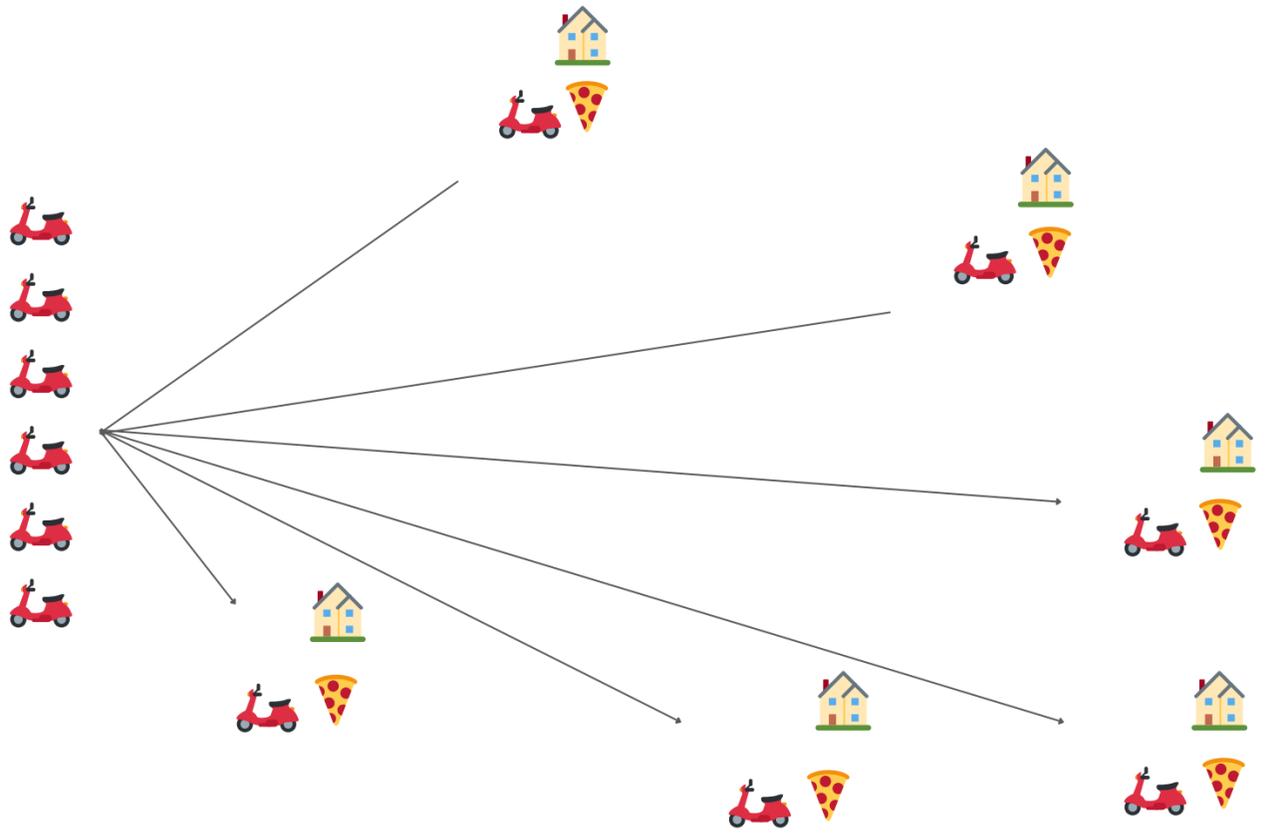




GPU vs CPU



CPU



GPU



Cloud Development Environment

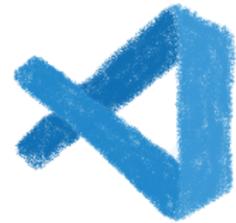
Cloud Development Environment



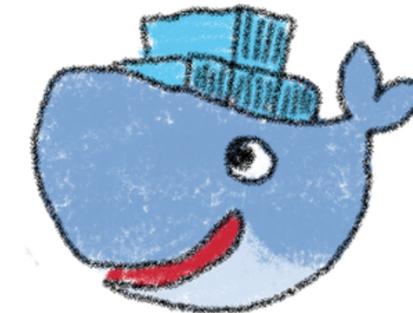
Le code du projet



Un éditeur de code



Un environnement d'exécution



À la demande

- Pour tester une idée
- Expérimenter différentes solutions
- Partager une session de travail entre collègues
- ... sky is the limit (et le coût de votre offre de CDE 😊)

Reproductible

- Environnement décrit précisément
- Configuration versionnée avec le code
- Cohérence entre l'environnement et le code lui même

Le CDE pour cet atelier : Gitpod



- Simplement ajouter <https://gitpod.io/#> devant l'URL de votre repository
- Deux fichiers 📄 de configuration :
 - .gitpod.dockerfile
 - .gitpod.yml
- Possibilité d'avoir par projets ou globales :
 - Clés SSH
 - variables d'environnement
- Tunneling possible avec le poste local



Python

101 pour cet atelier





Python

Pourquoi ce choix ?

-   Langage simple d'approche mais qui reste complet
-   Langage complet
-   Choix des communautés data science et data analysis



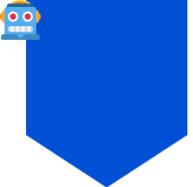


Python

Concepts pour aujourd'hui

-   Les fichiers requirements.txt
-   L'instruction import
-   Notebooks



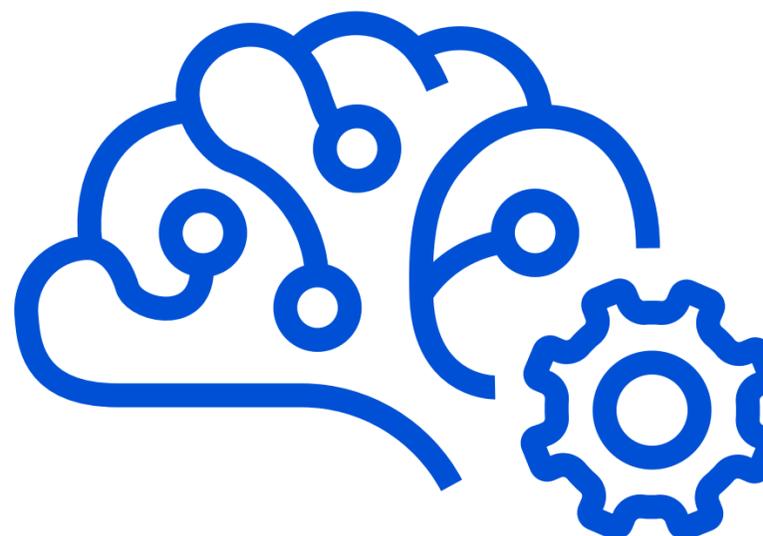


Faire de l'Intelligence Artificielle chez OVHcloud



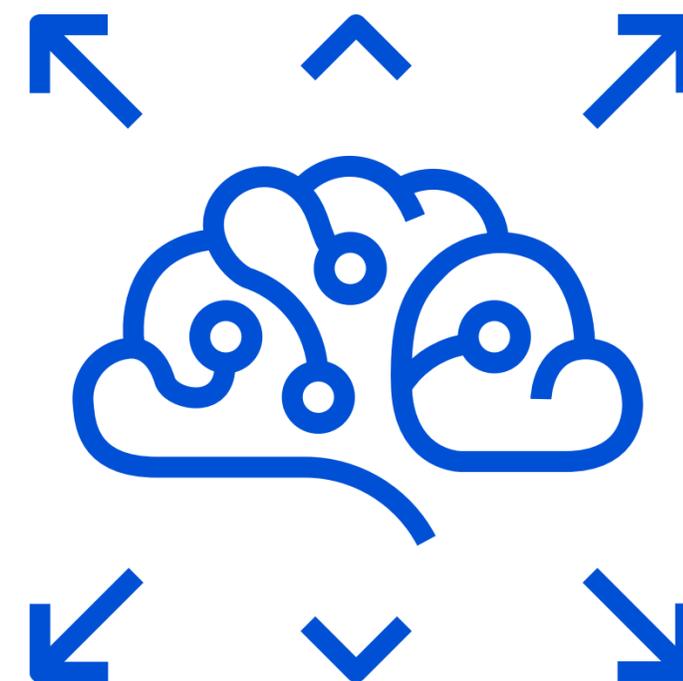
AI Notebooks

JupyterLab et VSCode,
images pré-construites



AI Training

GPU as a Service

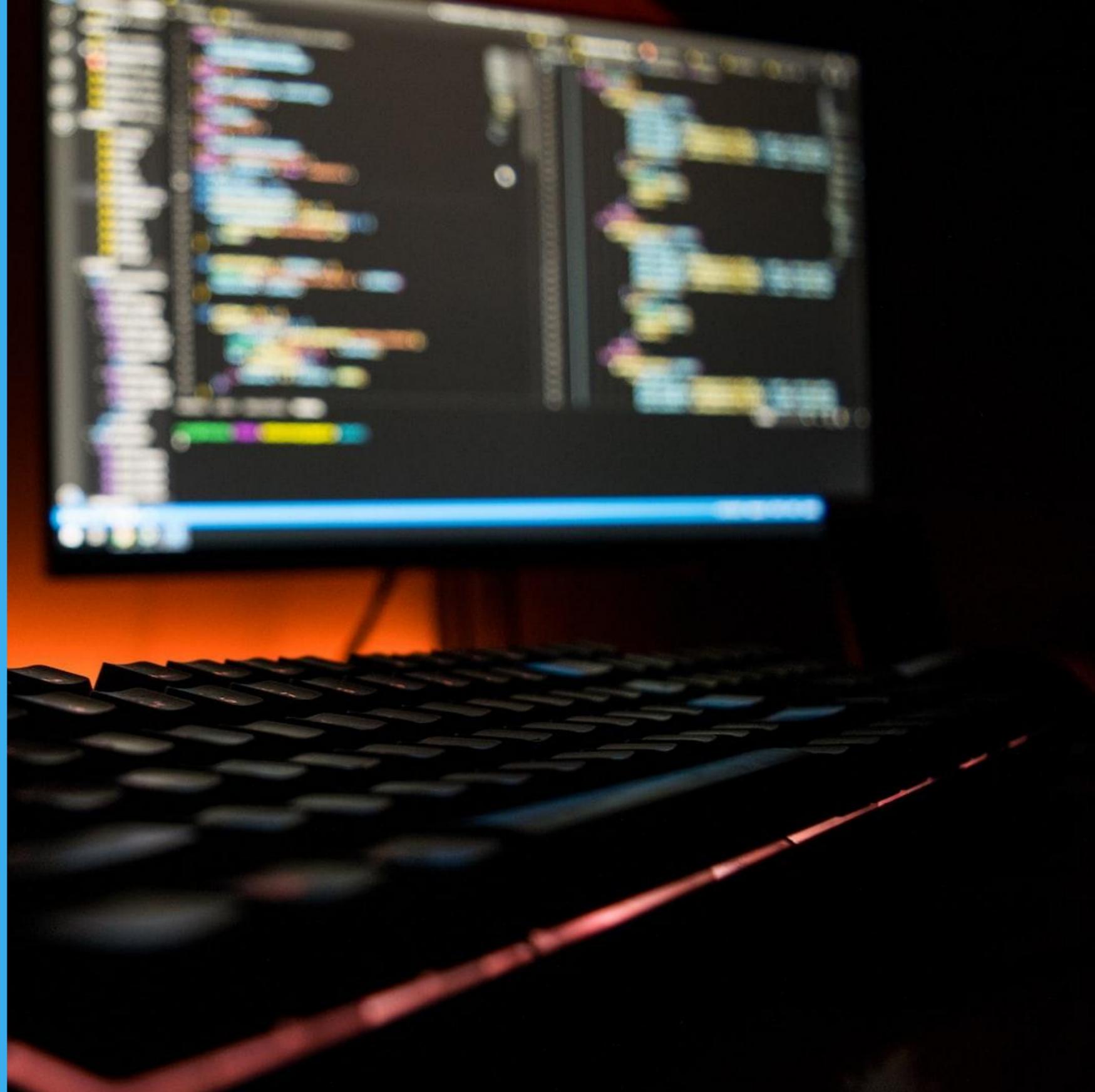


AI Deploy

CaaS pour l'IA



Public Cloud



Workshop Time !



Fork du repository GitHub

Projet à forker : <https://github.com/devrel-workshop/101-AI-and-py>

<https://ovh.to/ofPmni6>





Running

Opening Workspace ...

- devrelworksh-101aiandpy-ym...
<https://github.com/devrel-works...>

If you don't see an open dialog in your browser, make sure you have [VS Code](#) installed on your machine, and then click **Open in VS Code on Desktop** below.

More Actions... ▾

Open in VS Code on Desktop

i You can change the default editor for opening workspaces in [user preferences](#).

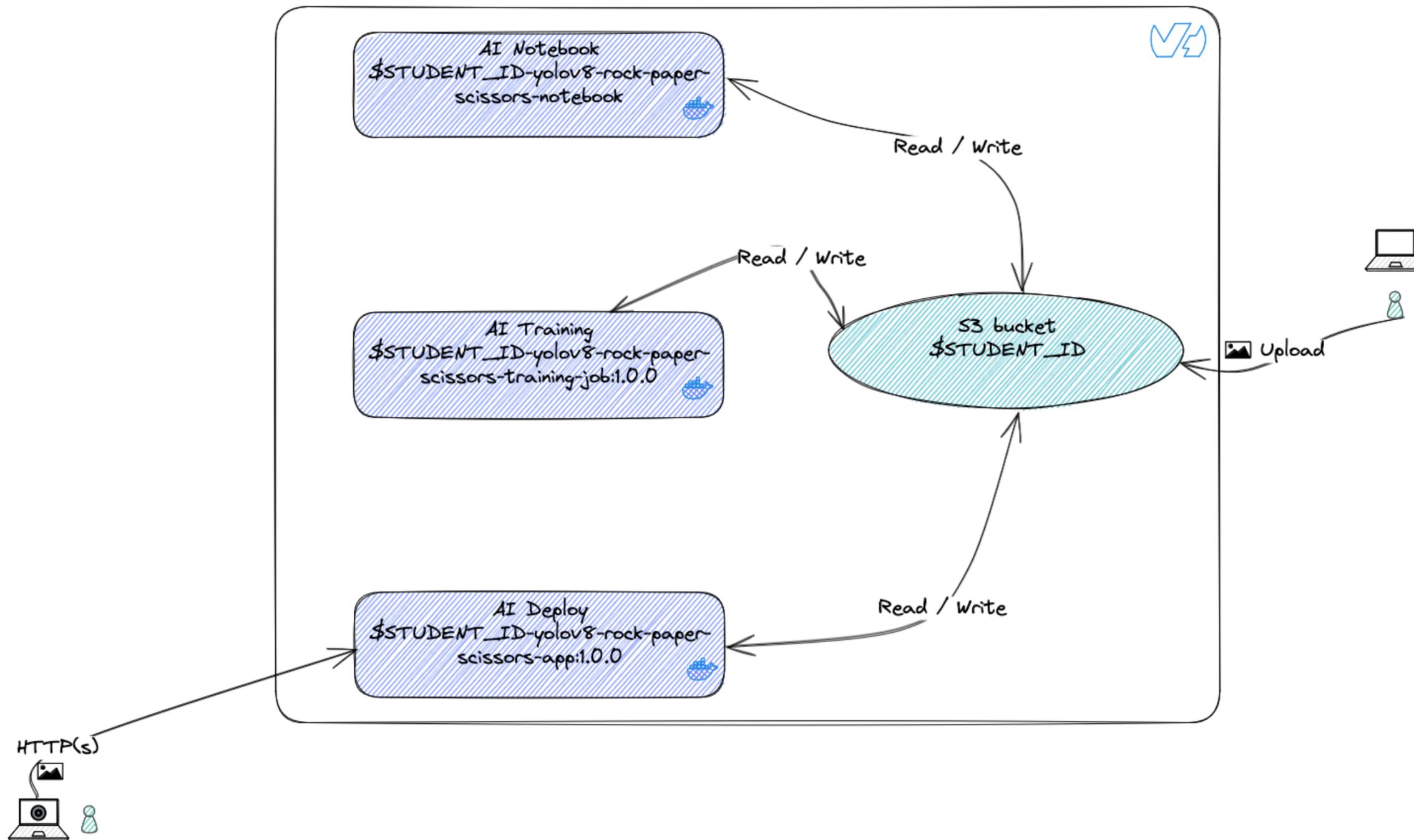
Démarrage de Gitpod



⚡ Initialisation de l'environnement Gitpod

- Créer un compte Gitpod (se connecter avec son compte GitHub)
- Prendre le modèle **large**
- Plus d'informations : section getting started du [README](#)

Architecture de l'application



Modèle avec AI Notebooks



AI Notebooks



Instructions

<https://github.com/devrel-workshop/101-AI-and-py/blob/main/docs/00-notebook.md>

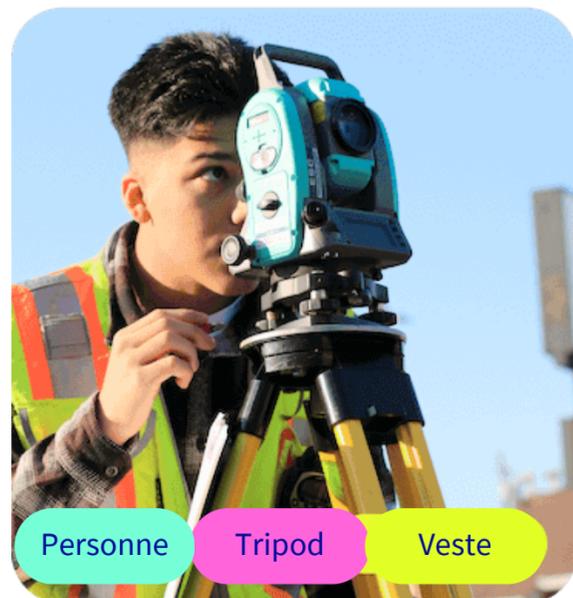
<https://ovh.to/Zz5AnB>



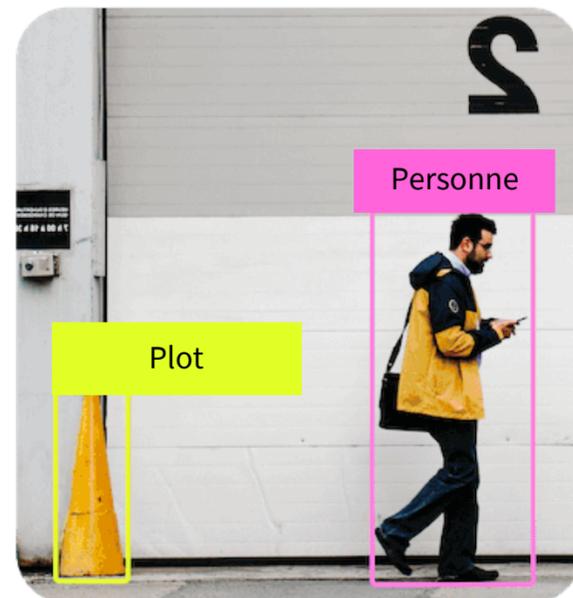


YOLOv8 (You only look once)

Classification



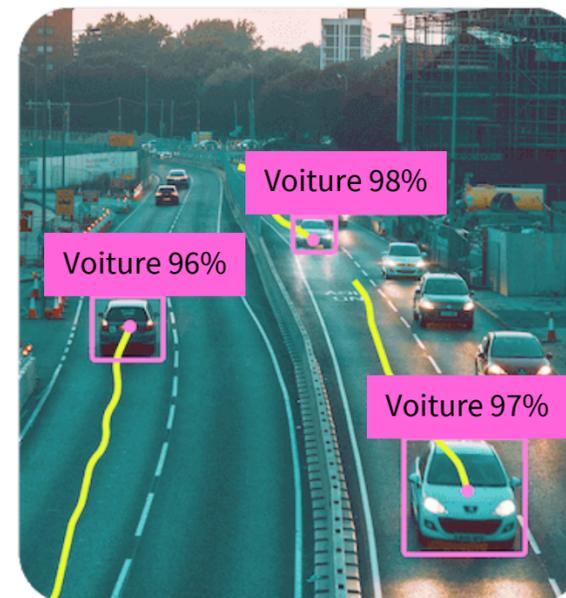
Détection



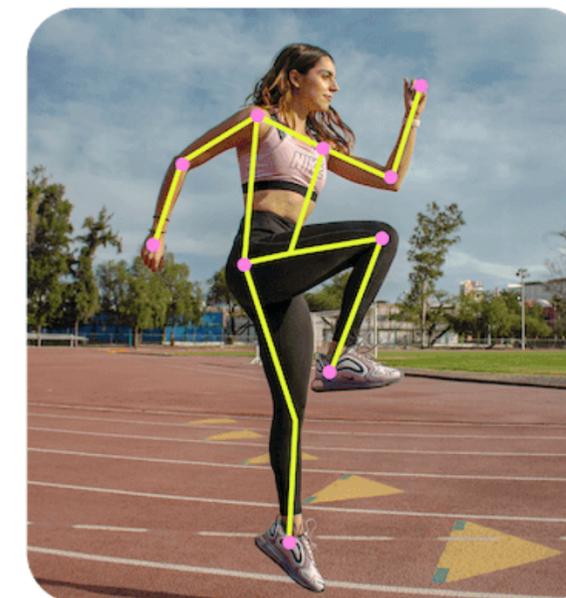
Segmentation



Suivi



Estimation de pose



<https://docs.ultralytics.com/> - <https://github.com/ultralytics/ultralytics>

Classification / Détection / Segmentation / Détection / Pose

Entraîné sur le dataset COCO (>200K images labellisées)

Lib python prête à l'emploi

Utilisation du plus petit modèle avec “seulement” 3.2 millions de paramètres



Un mot sur le Transfert Learning

- 🔄 Réutiliser un modèle déjà entraîné
- 🔍 Le spécialiser dans un domaine bien précis

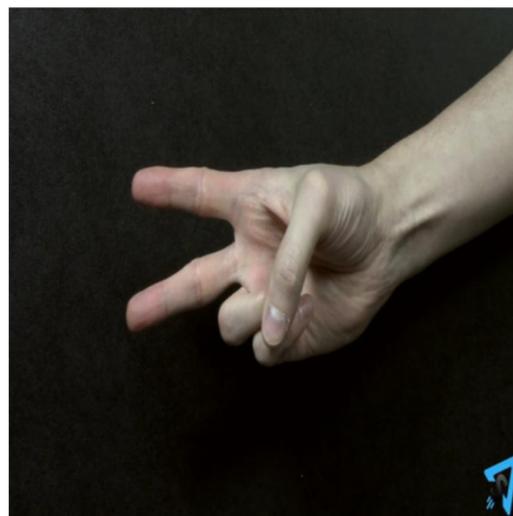
Exemple : détection d'objets divers qui devient une détection de signes



Le dataset utilisé



- Dataset Rock Paper Scissors SXSU récupéré de Roboflow
- Directement au bon format YOLOV8
- Pas de traitement sur les images / labels à faire
- +11 000 images
 - 10 953 pour l'entraînement (92%) (21% Papier, 29% Pierre, 20% Ciseaux)
 - 604 pour la validation (5.5%) (24% Papier, 25% Pierre, 20% Ciseaux)
 - 329 pour les tests (2.5%) (22% Papier, 22% Pierre, 21% Ciseaux)





Exécution du Notebook 1/2



- ▶ “Jouer” les cellules du Notebook (bouton ▶)
- ↳ Récupération du dataset
- ↳ Récupération des dépendances
- ✅ Vérification de la configuration matérielle (1 GPU) et logicielle (Ultralytics)
- ↳ Récupération du modèle [yolov8n](#) pré-entraîné sur COCO
- 🧪 Tester le modèle non entraîné avec une image de joueur



Exécution du Notebook 2/2



-  Entraîner le modèle avec les données du dataset “pierre / feuille / ciseaux”
-  Etude de la qualité du modèle
-  Tester le nouveau modèle, l'importance des “epochs”
-  Sauvegarder le modèle



⚡ Accès au Notebook

🔧 via le json *attende-conf.json* dans le workspace de projet Gitpod

```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai notebook list --token $AI_TOKEN
```

ID	NAME	STATE	AGE	FRAMEWORK	VERSION	EDITOR
xx-xx-xx-xx-xx	blabla_0	RUNNING	4h	conda	conda-py311-cudaDevel11.8	jupyterlab

```
URL
https://xx-xx-xx-xx-xx.notebook.bhs.ai.cloud.ovh.net
```

Analyse des résultats

Vrai Positif (TP): Détection correcte par le modèle (Objet x présent dans l'image détecté comme x, avec sa détection dans la même boîte englobante que la groundtruth)



Vérité



Prédiction

Faux Positif (FP): Détection incorrecte par le modèle (Objet x présent dans l'image détecté comme y).



Vérité



Prédiction

Faux Négatif (FN): Une Ground-truth de loupée par le modèle (pas détectée) (Objet x non détecté alors qu'il est présent dans l'image).

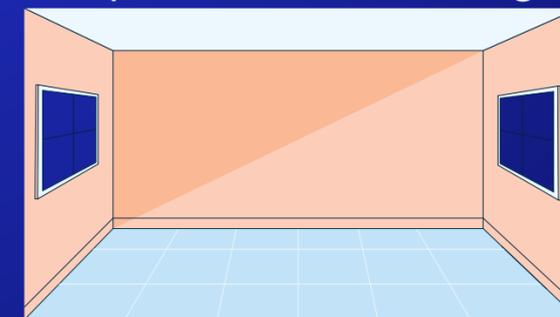


Vérité

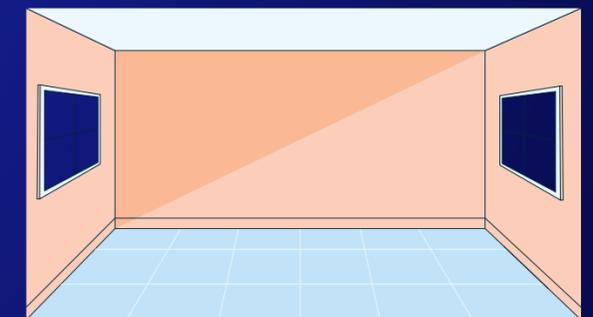


Prédiction

Vrai Négatif (TN): Images restantes, où aucune détection n'a eu lieu parce qu'il n'y avait pas le signe en question dans l'image.



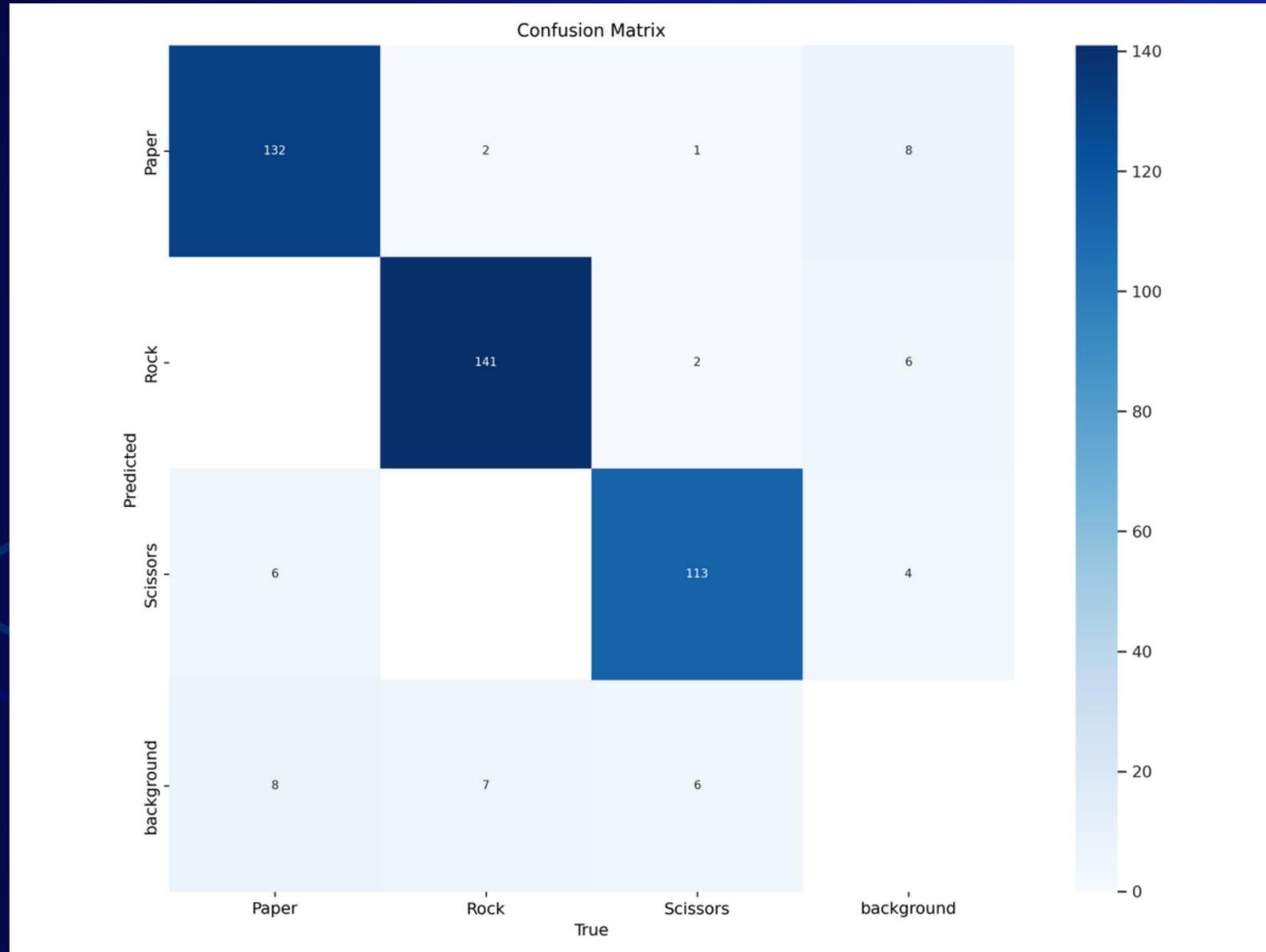
Vérité



Prédiction



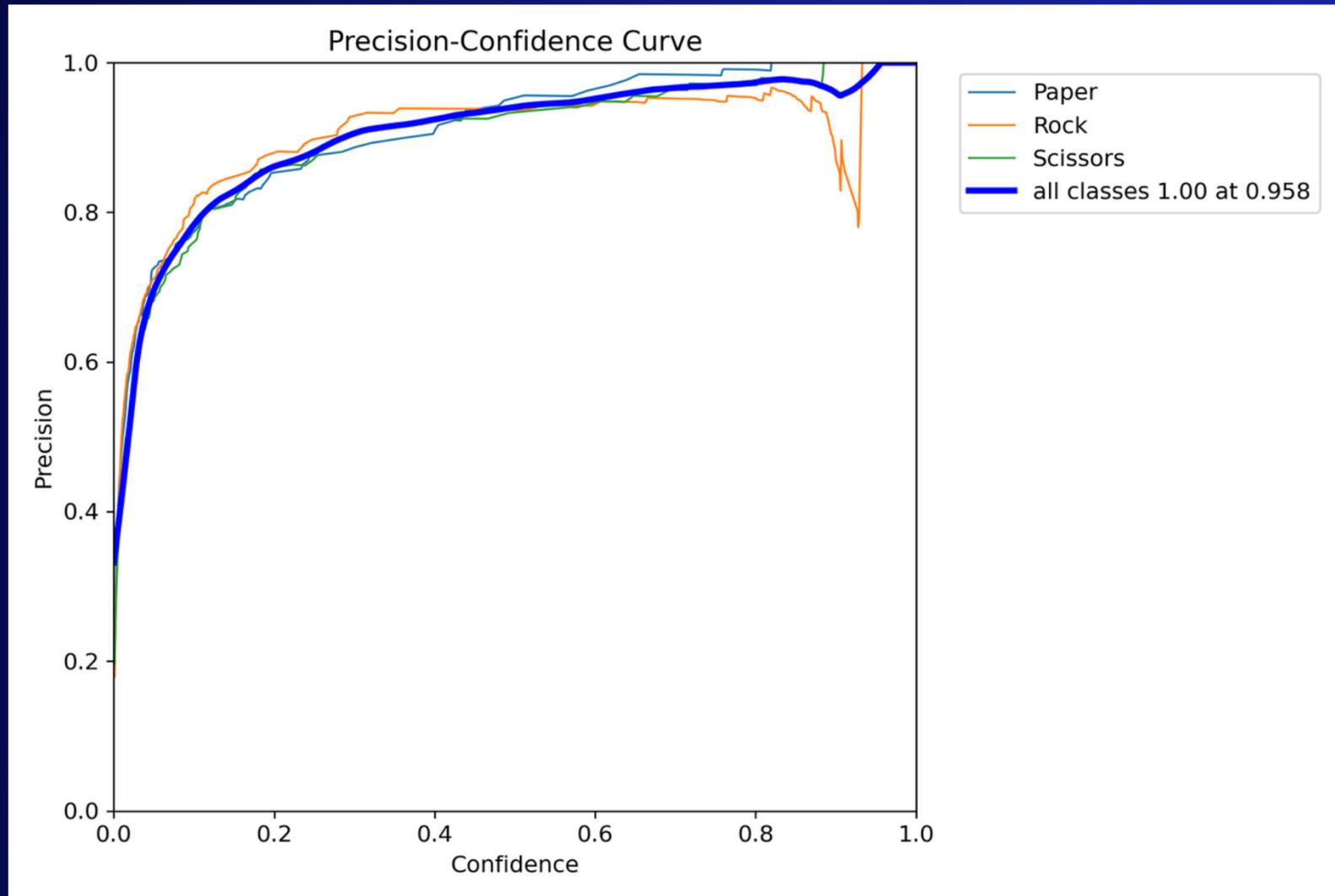
Analyse des résultats



- Vue résumée des classifications (Nombre de TP, TN, FP, FN)
- La diagonale représente les bonnes prédictions, tandis que les autres sont des erreurs.
- Classe Background

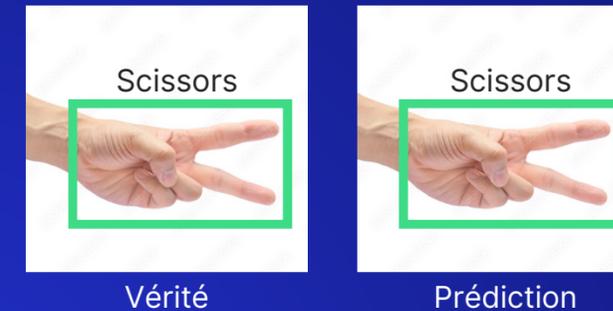
Matrice de confusion

Précision

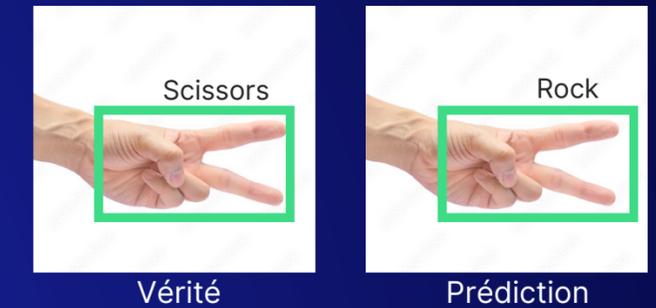


Seuil de confiance : Détermine le score de confiance minimum requis pour qu'une prédiction soit considérée comme valide

Vrai Positif (TP)



Faux Positif (FP):



$$\text{Précision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

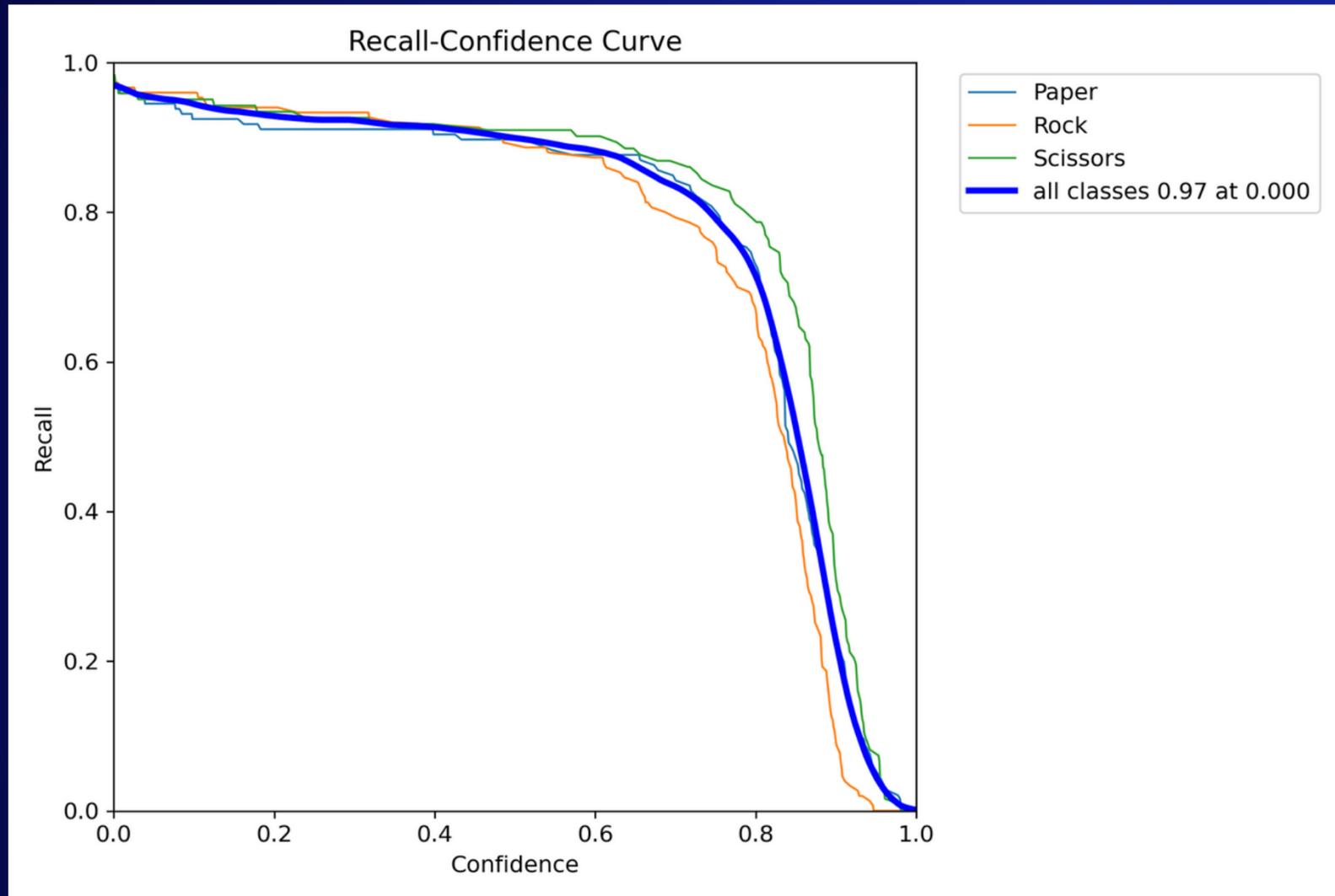
Plus le seuil de confiance sera bas, plus on aura de prédictions, avec de nombreuses erreurs (FP) -> la précision sera basse

Mais attention à ne pas en fixer un trop haut non plus (très peu de TP détectés)

Elle est utile lorsque le coût d'un FP est élevé



Rappel (Recall)



Seuil de confiance : Détermine le score de confiance minimum requis pour qu'une prédiction soit considérée comme valide

Vrai Positif (TP)



Vérité



Prédiction

Faux Négatif (FN)



Vérité



Prédiction

$$\text{Rappel} = \frac{\text{TP}}{\text{TP+FN}}$$

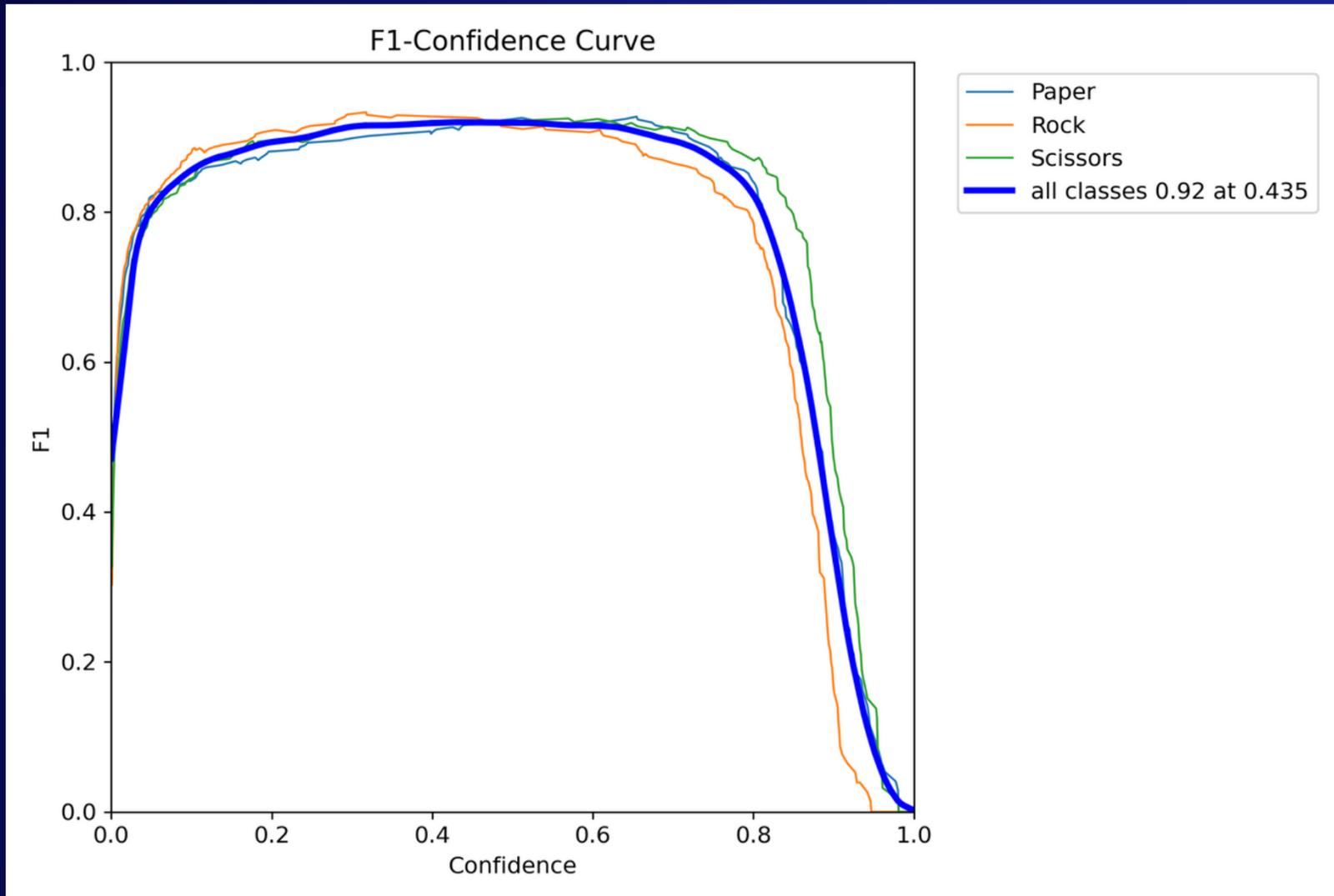
Nombre de fois où le modèle a correctement prédit un exemple positif sur le nombre total d'exemples positifs.

Il est utile lorsque le coût d'un FN est élevé

Compromis entre la précision et le rappel



Score F1

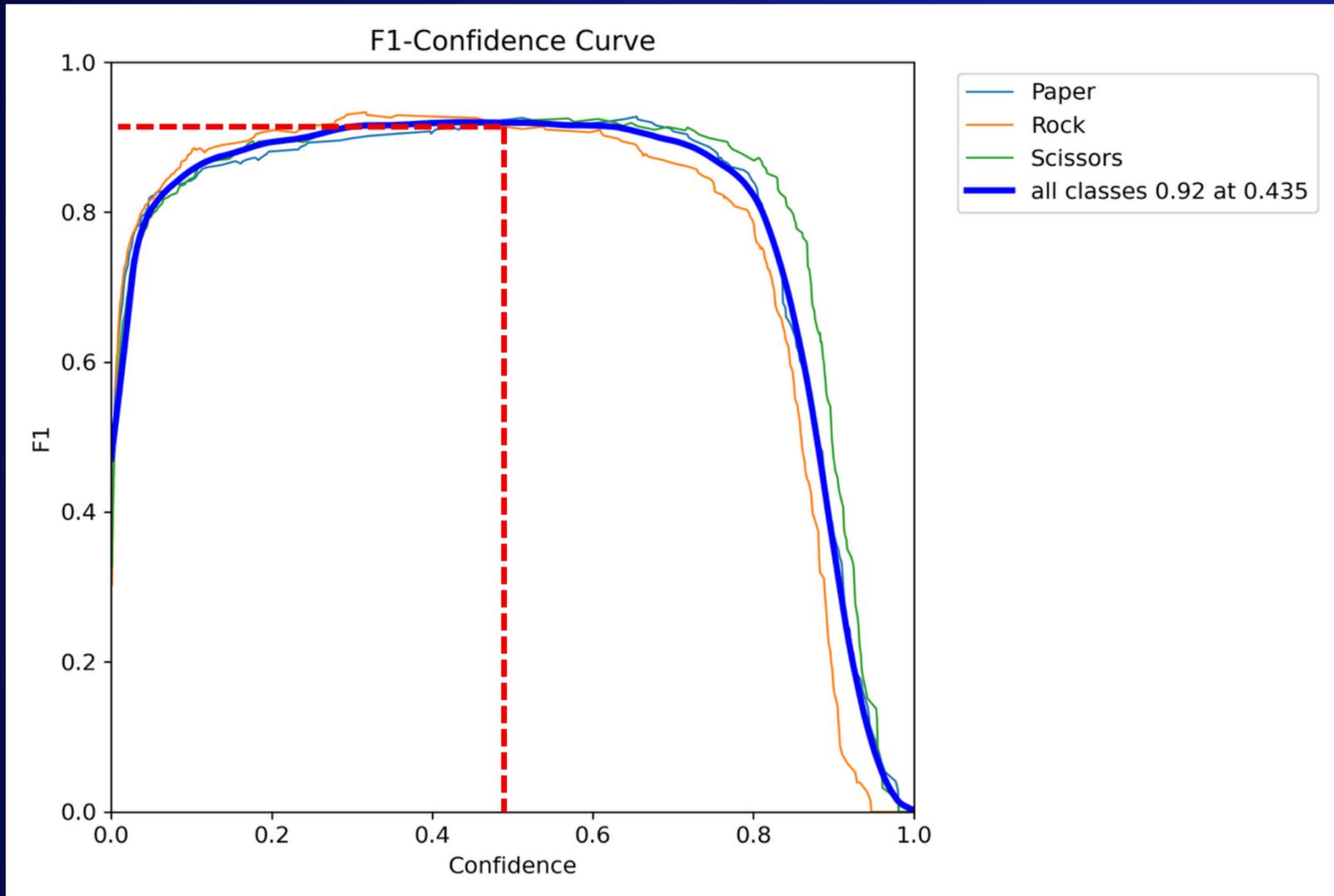


Une combinaison de la précision et du rappel en une seule et même métrique

Seuil de confiance : Détermine le score de confiance minimum requis pour qu'une prédiction soit considérée comme valide



Score F1

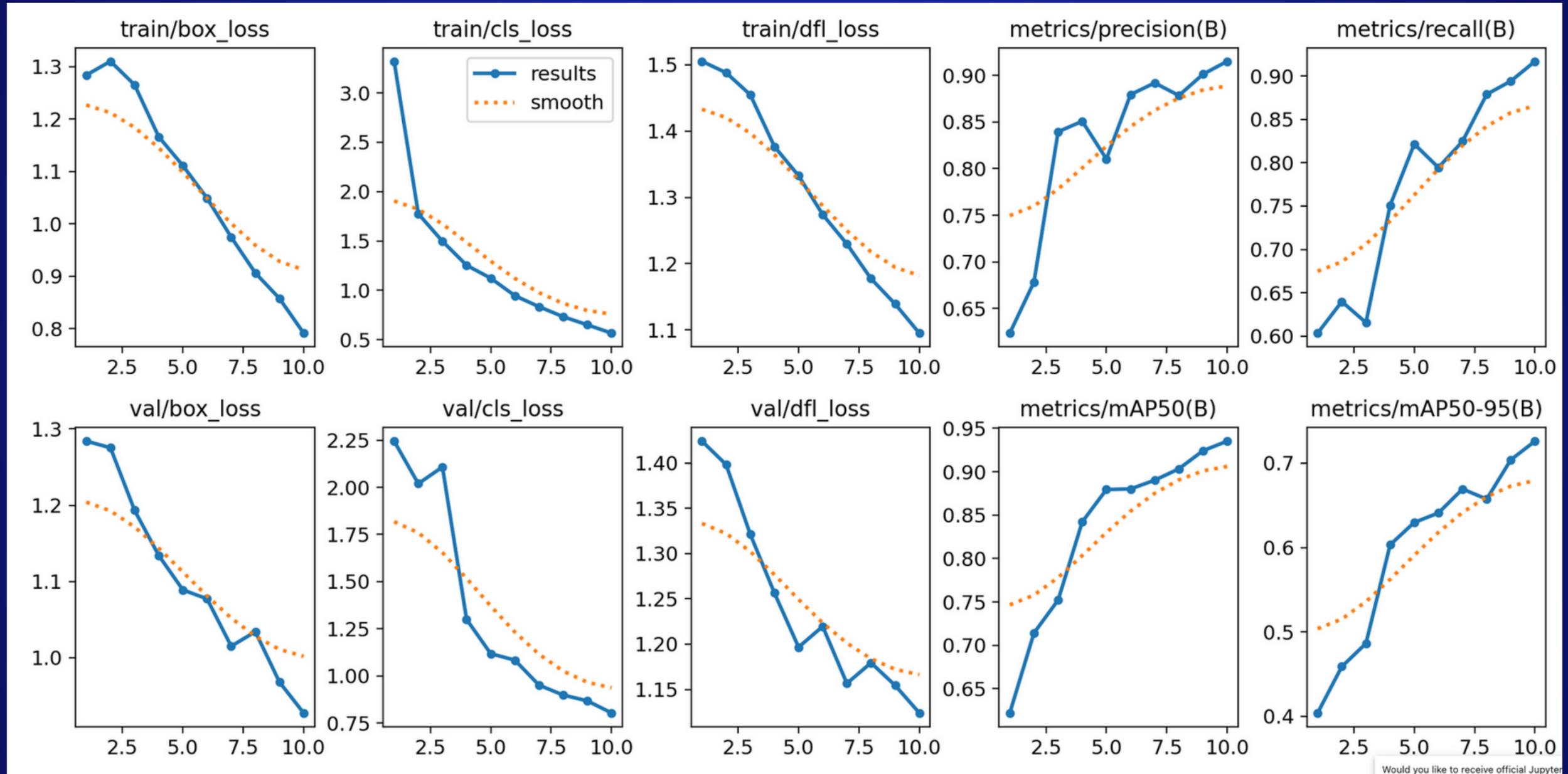


Une combinaison de la précision et du rappel en une seul et même métrique

Seuil de confiance : Détermine le score de confiance minimum requis pour qu'une prédiction soit considérée comme valide

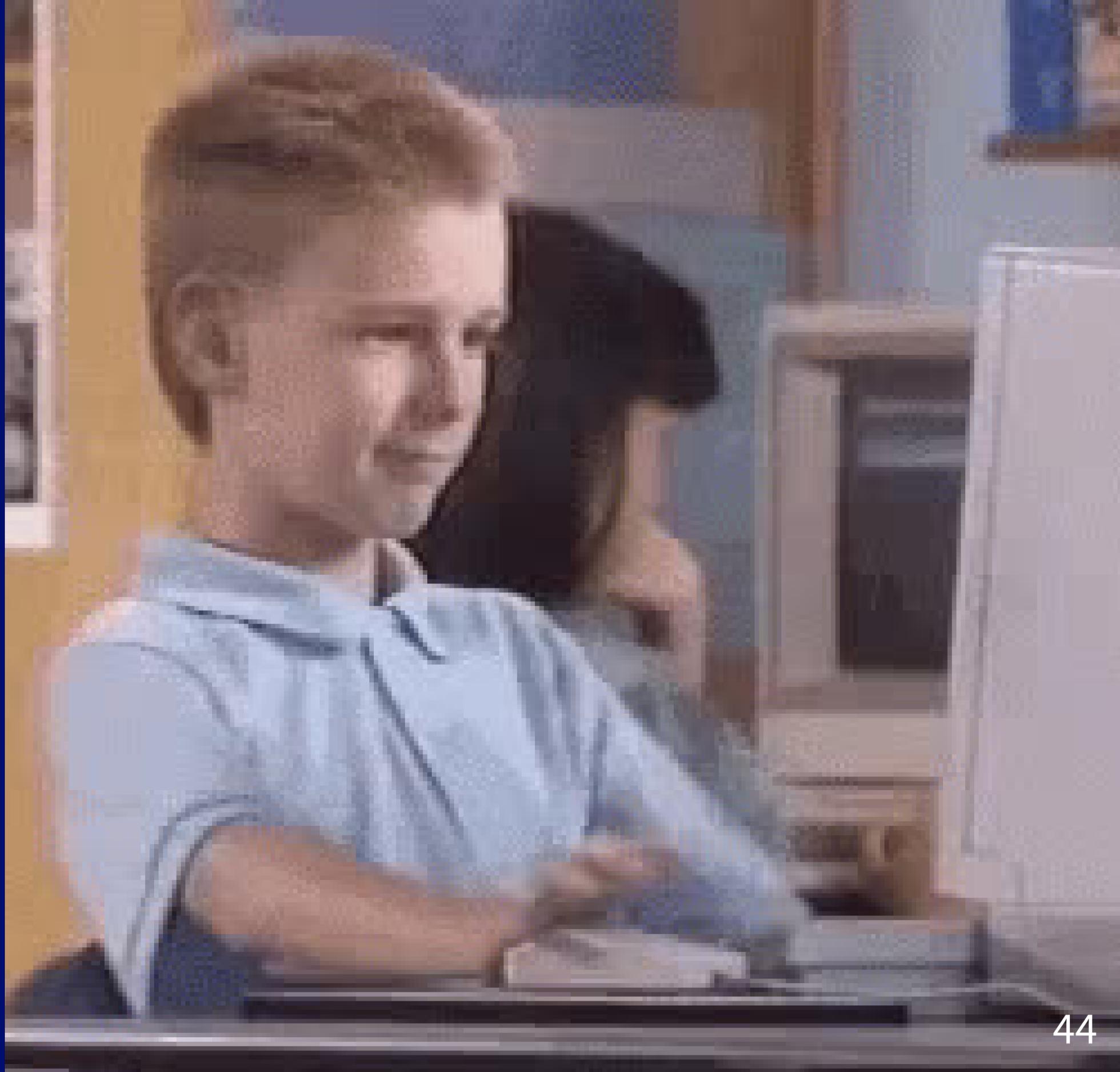


Analyse des résultats



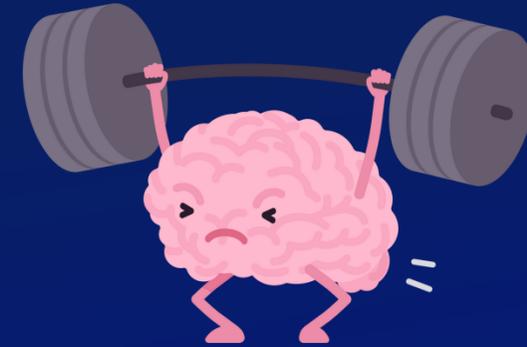
 **Félicitations** 

Vous avez créé votre
premier modèle
d'intelligence
artificielle !





AI Training



Entraînement avec AI Training



Instructions

<https://github.com/devrel-workshop/101-AI-and-py/blob/main/docs/01-training.md>

<https://ovh.to/tFHguV>





i Rappels

- C'est le même token que celui du Notebook
- C'est le même object storage que celui du Notebook



Création de l'image

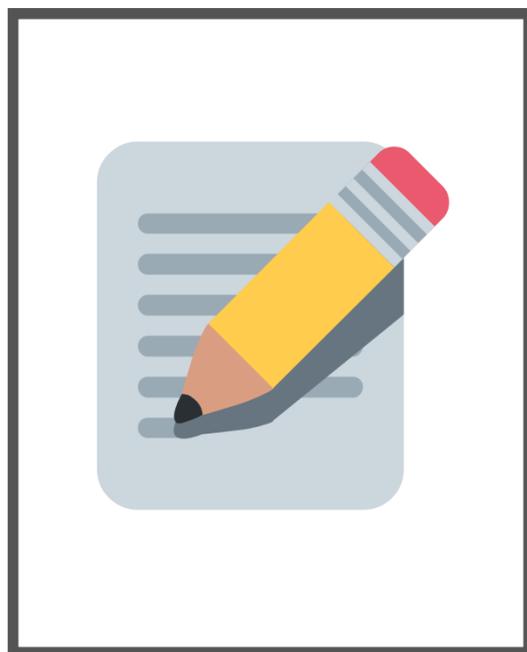
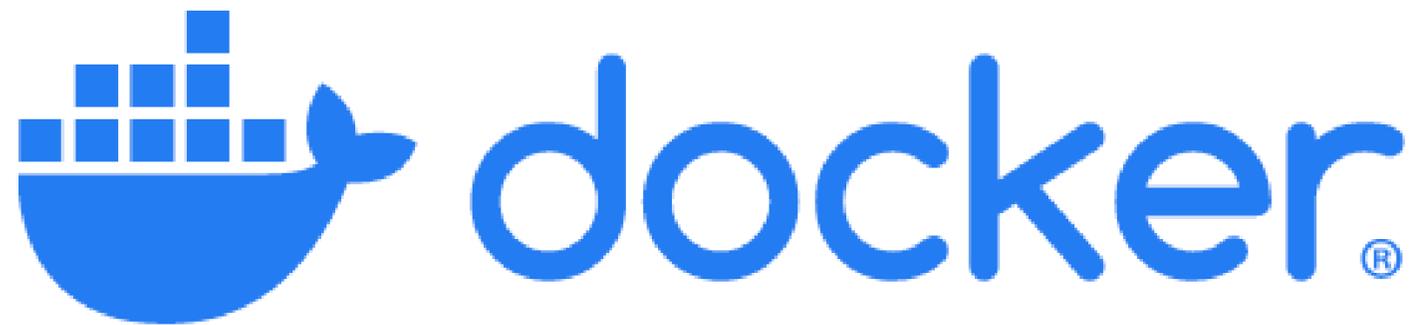
📁 Répertoire de travail : **src/training**

- **Dockerfile**: le dockerfile pour construire l'image
- **Requirements.txt** : fichier de gestion des dépendances Python
- **Train.py** : script Python pour l'entraînement du modèle

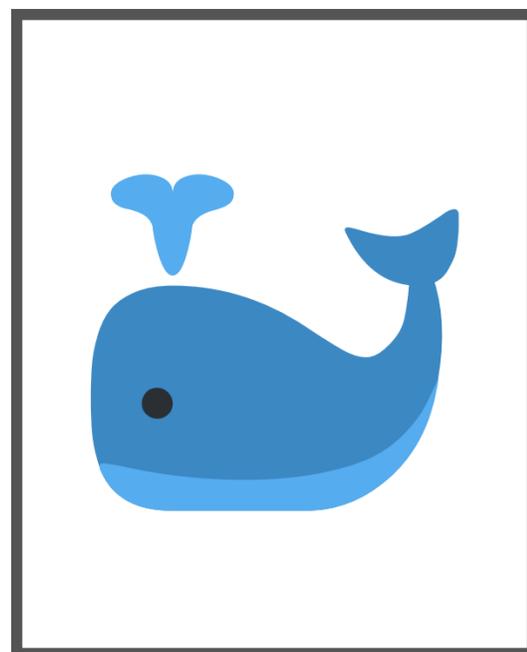
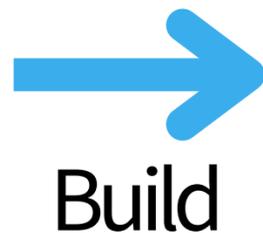
🐳 Fabrication de l'image

🐛 (Optionnel) Run / debug localement

⬆️ Push de l'image dans la registry : <user>/ <pass>



Docker File



Docker Image

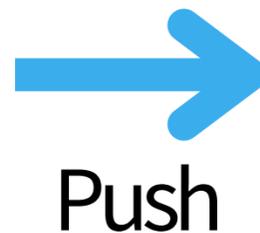
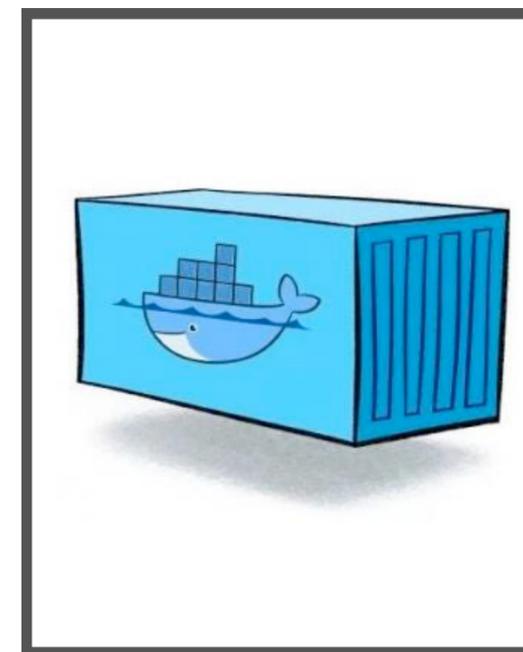
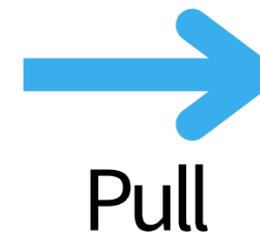


Image Registry



Docker Container

⚡ Création du Job avec la CLI



```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai job run \
> --token $AI_TOKEN \
> --name $STUDENT_ID-yolov8-rock-paper-scissors-training-job \
> --gpu 1 \
> --env NB_OF_EPOCHS=10 \
> --volume $STUDENT_ID@S3GRA:/workspace/attende:RW:cache \
> --unsecure-http \
> $REGISTRY_NAME/$STUDENT_ID/yolov8-rock-paper-scissors-training-job:1.0.0
```

Plan B



```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai job run \
> --token $AI_TOKEN \
> --name $STUDENT_ID-yolov8-rock-paper-scissors-training-job \
> --gpu 1 \
> --env NB_OF_EPOCHS=10 \
> --volume backup-0@S3GRA:/workspace/attendee:RW:cache \
> --unsecure-http \
> $REGISTRY_NAME/backup-0/yolov8-rock-paper-scissors-training-job:1.0.0
```

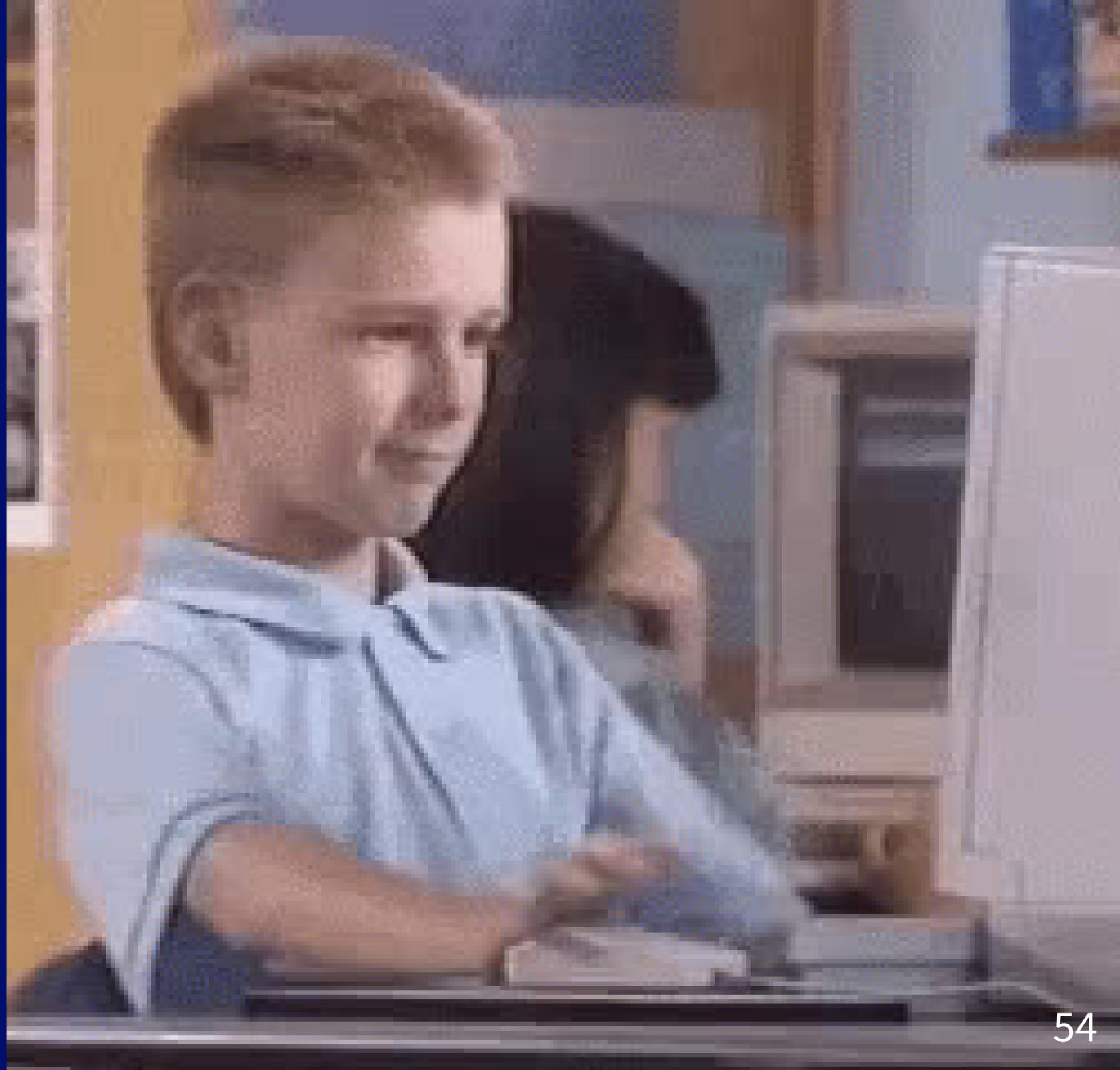


Suivi des logs

```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai job logs -f <job id> --token $AI_TOKEN
```

 **Félicitations** 

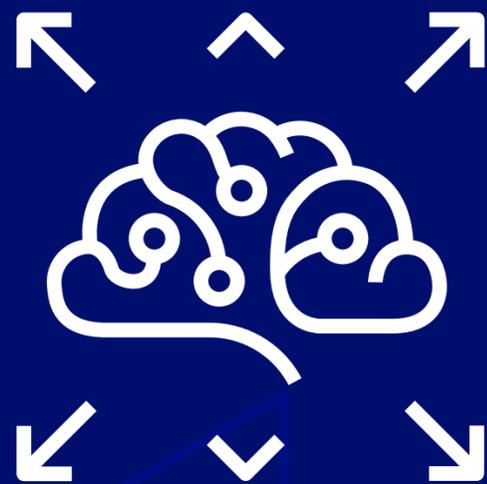
Vous avez entraîné
votre premier modèle
d'intelligence
artificielle !



Application

avec

AI Deploy



Welcome on the 🗑️ 📄 ✂️ game!

Take your picture in real time:



Take Photo



i Rappels

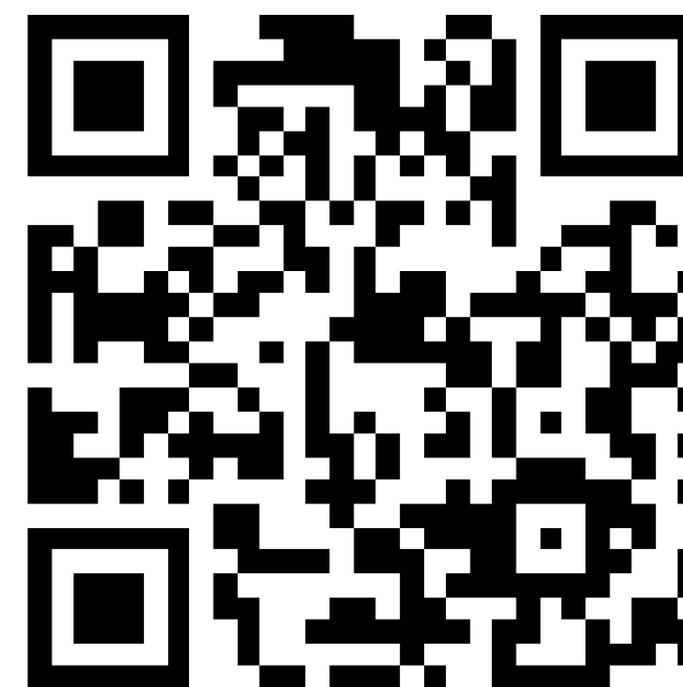
- C'est le même token que celui du Notebook
- C'est le même object storage que celui du Notebook



Instructions

<https://github.com/devrel-workshop/101-AI-and-py/blob/main/docs/02-application.md>

<https://ovh.to/DGoWaJ>





Création de l'image

📁 Répertoire de travail : **src/app**

- **Dockerfile** : le dockerfile pour construire l'image
- **Requirements.txt** : fichier de gestion des dépendances Python
- **App.py** : script Python pour la création de l'application

🗑️ Si nécessaire supprimer les images d'entraînement

🐳 Fabrication de l'image

🐛 (Optionnel) Run / debug localement

⬆️ Push de l'image dans la registry



⚡ Création de l'application avec la CLI

```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai app run \
> --token $AI_TOKEN \
> --name $STUDENT_ID-yolov8-rock-paper-scissors-app \
> --cpu 1 \
> --default-http-port 8501 \
> --volume $STUDENT_ID@S3GRA:/workspace/attende:RW:cache \
> --unsecure-http \
> $REGISTRY_NAME/$STUDENT_ID/yolov8-rock-paper-scissors-app:1.0.0
```

Plan B



```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai app run \
> --token $AI_TOKEN \
> --name $STUDENT_ID-yolov8-rock-paper-scissors-app \
> --cpu 1 \
> --default-http-port 8501 \
> --volume backup-0@S3GRA:/workspace/attendee:RW:cache \
> --unsecure-http \
> $REGISTRY_NAME/backup-0/yolov8-rock-paper-scissors-app:1.0.0
```



Suivi des logs

```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai app logs -f <app id> --token $AI_TOKEN
```

Accéder à l'application



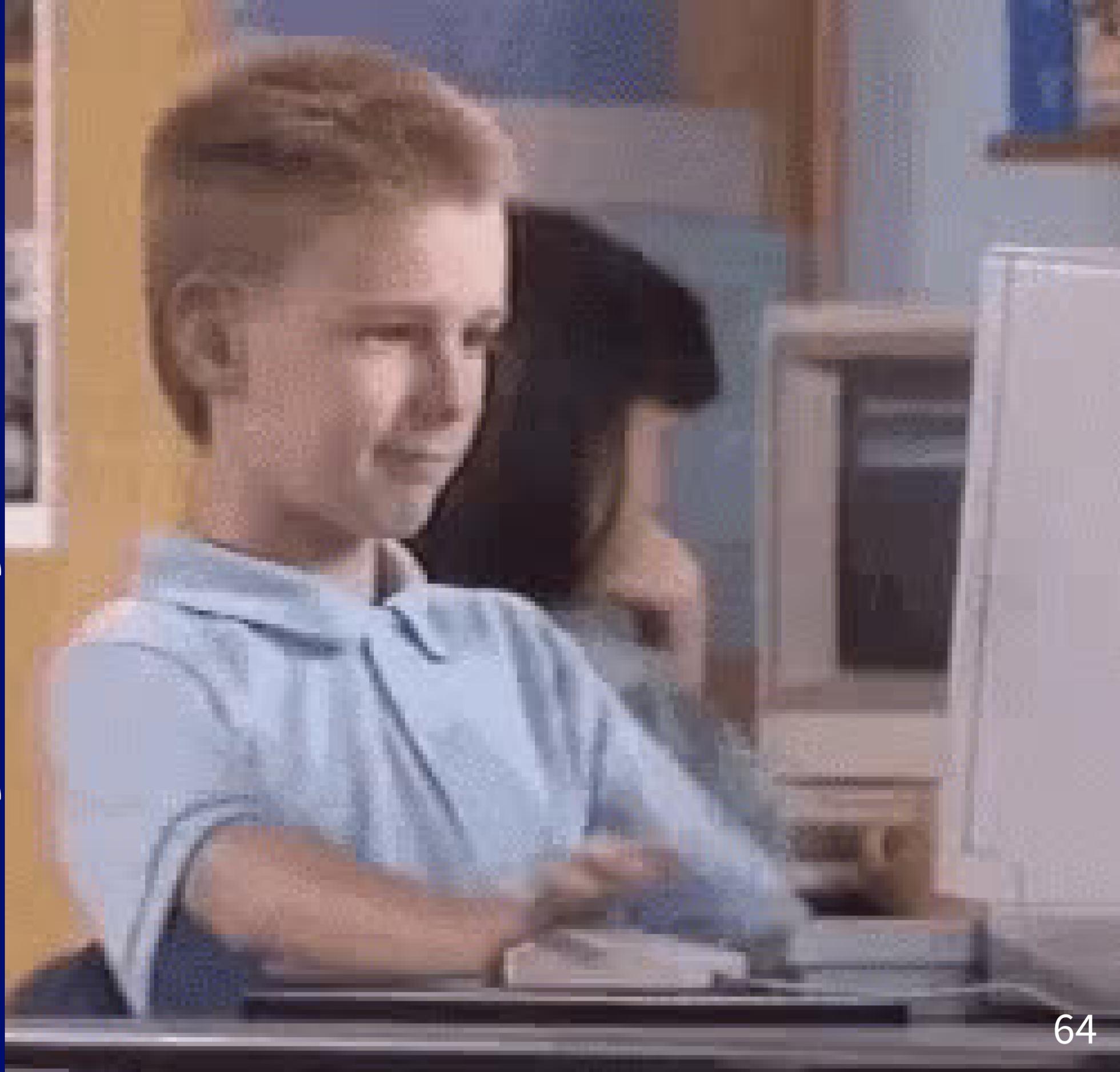
```
OVHcloud Instance
ubuntu@my-instance:~$ ovhai app get <Appld> --token $AI_TOKEN

Status:
State:          SCALING
Internal Service Ip:  ~
Available Replicas:  0
Url:            https://<Appld>.app.gra.ai.cloud.ovh.net
Grpc Address:    <Appld>.app-grpc.gra.ai.cloud.ovh.net:443
Info Url:       https://ui.gra.ai.cloud.ovh.net/app/<Appld>
Monitoring Url: https://monitoring.gra.ai.cloud.ovh.net/d/app?var-app=<Appld>&from=1704720216889
```

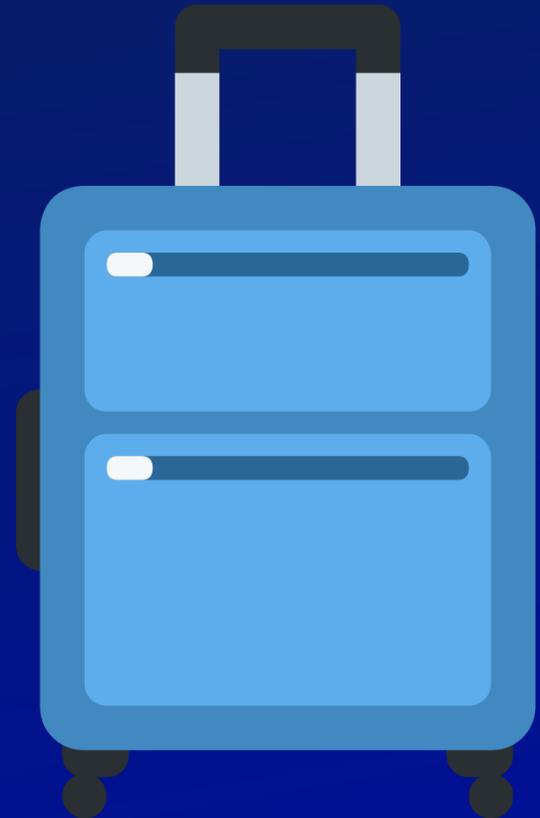
 **Félicitations** 

Vous avez créé et
déployé votre première
application basée sur
l'intelligence artificielle

!



**Take away /
Next ?**



Un Workflow typique en IA

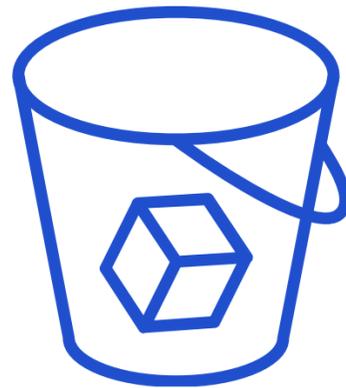
STOCKER SA DATA



- 📀 Il faut une très grande quantité de données
- 🔍 Il existe des datasets pré-crés (gratuits ou payants)
- 📀 La plupart du temps la donnée est brute
- ⚠️ Attention aux coûts (stockage, lecture / écriture)

Un Workflow typique en IA

STOCKER SA DATA



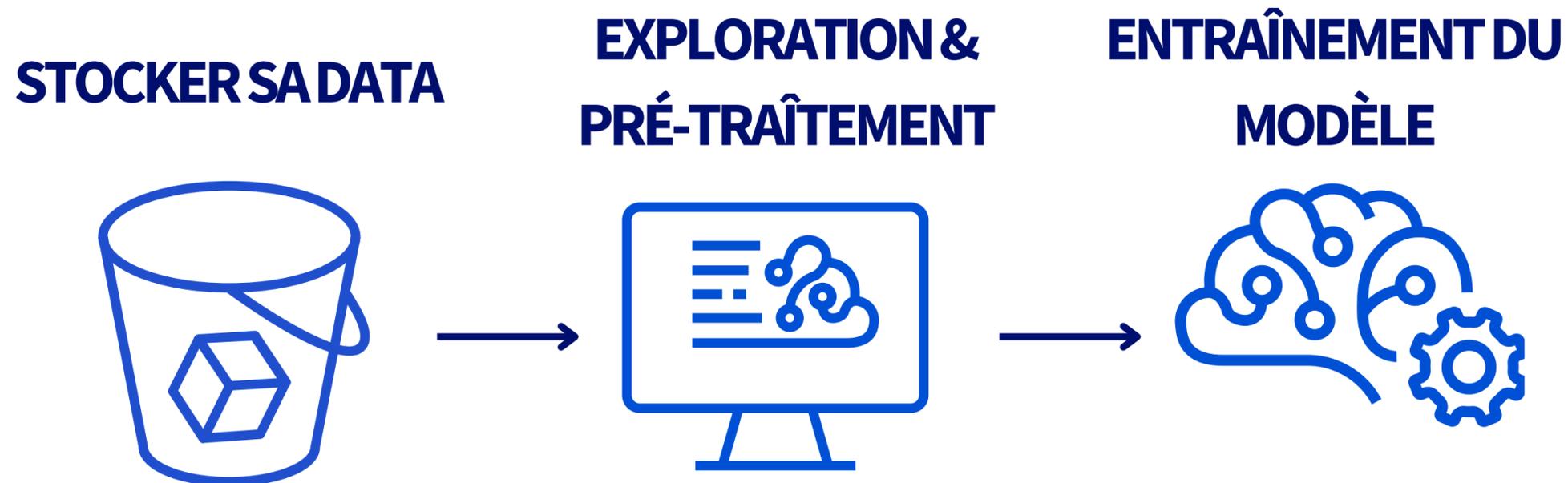
EXPLORATION &
PRÉ-TRAÎTEMENT



🔧 Créer des datasets nettoyés (Valeurs manquantes, Normalisation, ...)

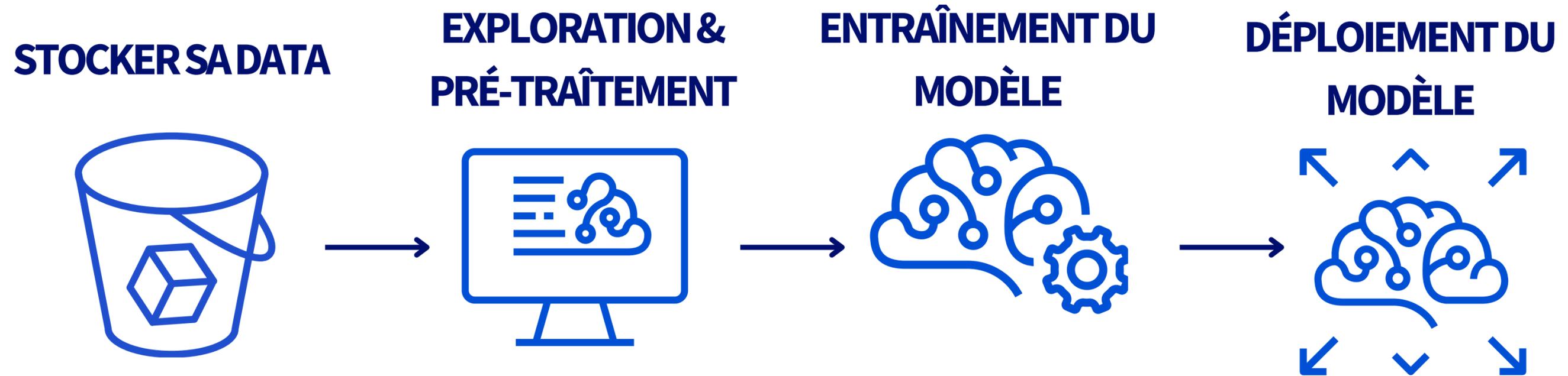
🎯 Le but est de traduire les données brutes dans un langage compréhensible par le modèle

Un Workflow typique en IA



- 🧠 Le “cerveau” de l’application
- 📊 Basé sur des opérations mathématiques complexes
- 🌟 Permet d’optimiser le modèle
- 📀 Il faut une grande quantité de données
- 🔋 Nécessite de la puissance de calcul
- 💰 Il existe des modèles pré-entraînés (payants ou gratuits)

Un Workflow typique en IA



-  Utilise le modèle pour faire l'inférence
-  Peut exposer une API ou une interface utilisateur
-  Haute disponibilité



Quels sont les métiers?

Data Scientist

ML Engineer

Dev



LA OU LE DATA SCIENTIST



- 🔍 Analyse et manipulation des données
- 💻 Développement
- 🧠 Machine Learning
- 📊 Mathématiques
- % Statistiques

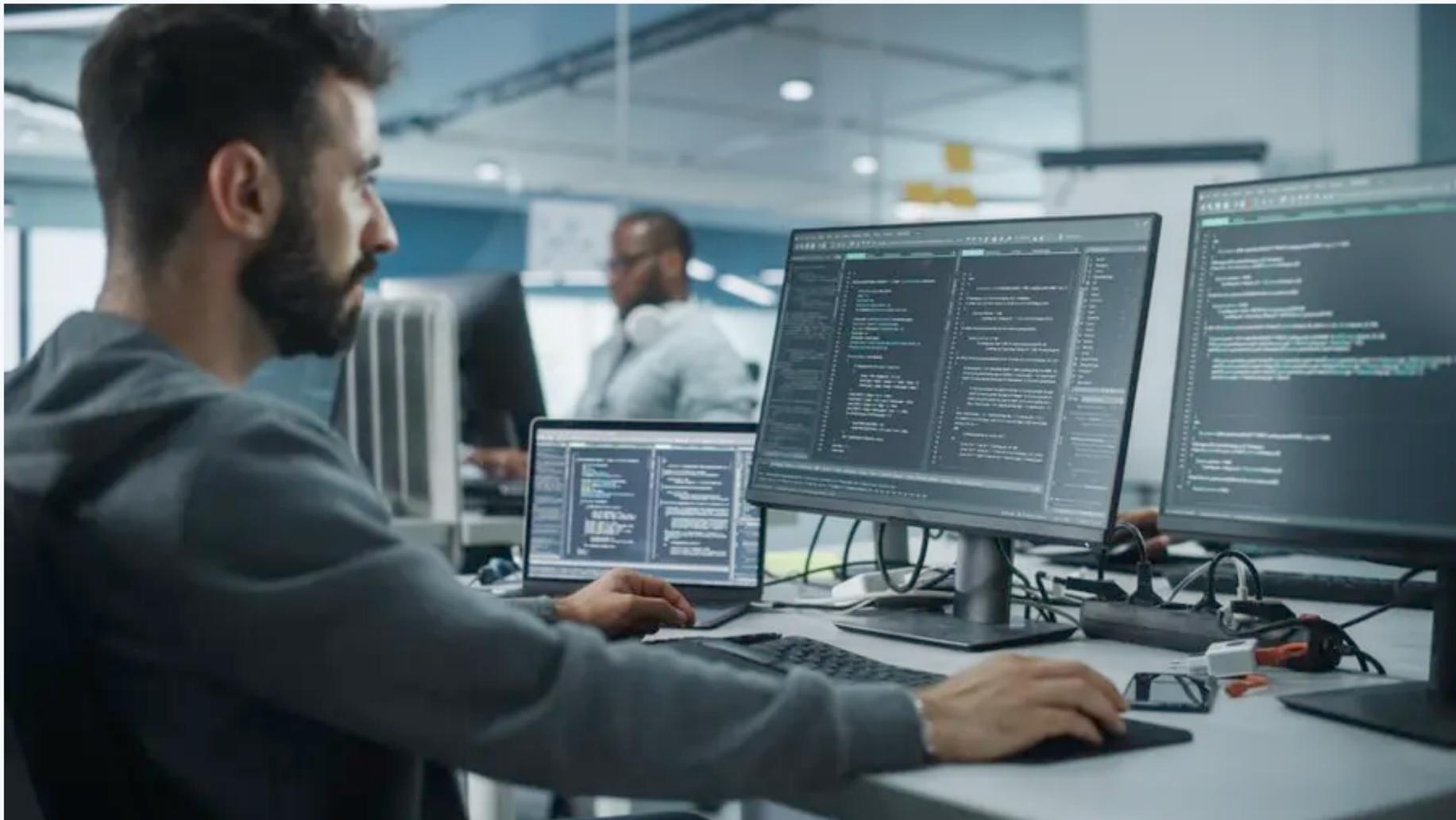


LA OU LE MACHINE LEARNING ENGINEER

- 🧑‍🔬 Data Science
- 📊 Mathématiques
- % Statistiques
- 💻 Développement
- 🧰 Frameworks & outillages IA
- ☁️ Cloud et conteneurisation
- 👷 Ops



LA DÉVELOPPEUSE OU LE DÉVELOPPEUR



-  Python
-  Concepts IA
-  Développement
-  Frameworks et outillages IA
-  Cloud et conteneurisation



Goh Rhy Yan

⚠️ LES BIAIS

- 👤 LES IA SONT CRÉÉES PAR DES HUMAINS
- 💿 LES DONNÉES SONT DE PLUS OU MOINS BONNE QUALITÉ
- 🏷️ LA LABELLISATION DES DONNÉES EST SOUVENT FAITE PAR DES HUMAINS



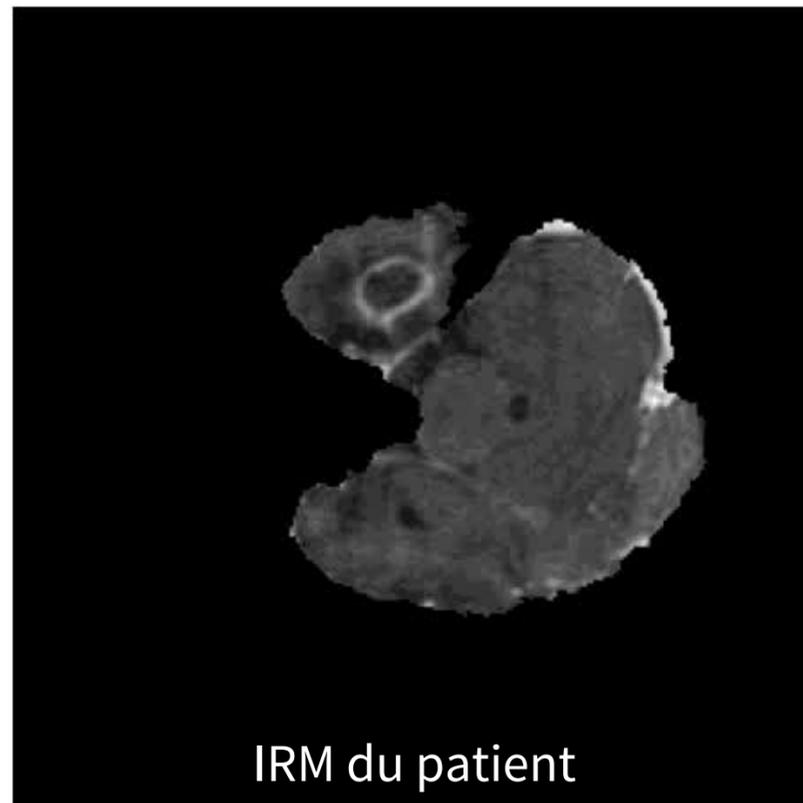
“Génère moi une image de médecins”

L'IA N'EST PAS SOURCE DE VÉRITÉ

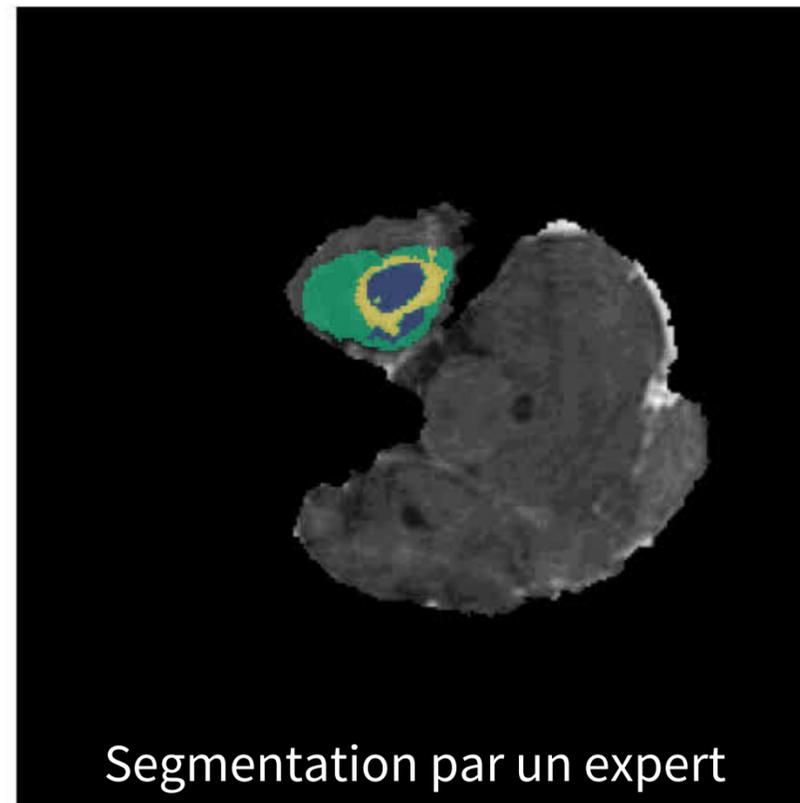
% CELA NE RESTE QUE DES PROBABILITÉS

🤔 C'EST UNE ESTIMATION

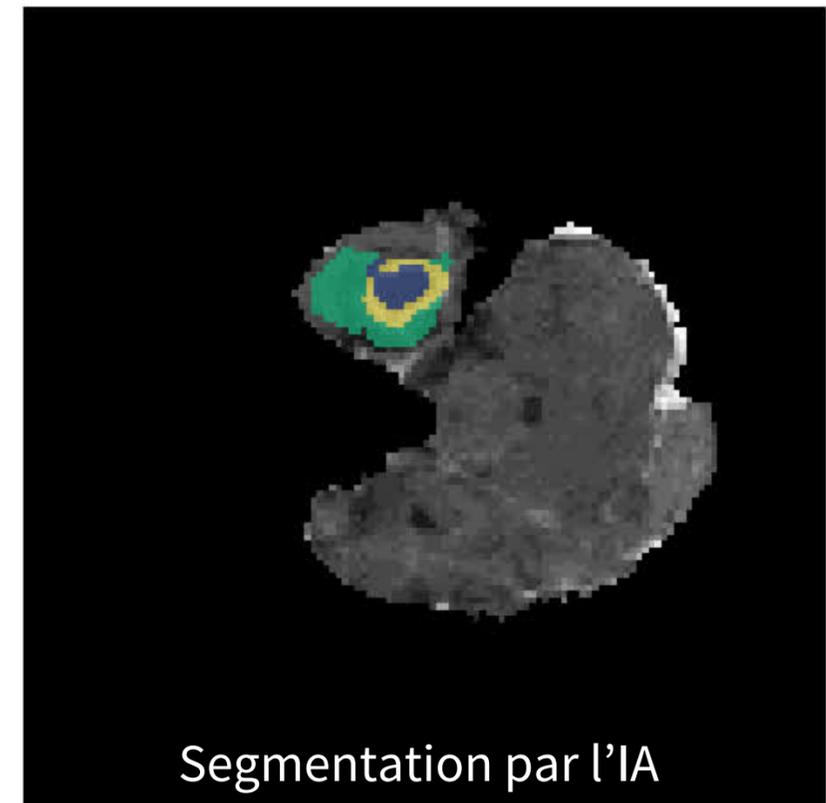
🤖 CELA N'EST QU'UNE AIDE À LA DÉCISION ET NE REMPLACE PAS UN·E HUMAIN·E



IRM du patient



Segmentation par un expert



Segmentation par l'IA

LA COURSE À LA PUISSANCE

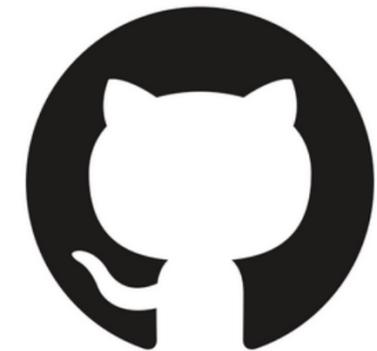
-  TOUJOURS PLUS DE DONNÉES
-  DE MODÈLES TOUJOURS PLUS GROS AVEC PLUS DE COUCHES
-  PLUS DE PUISSANCE DE CALCUL
-  DES CONSOMMATIONS QUI AUGMENTENT
-  UTILISER L'IA DE MANIÈRE "INTELLIGENTE"
-  AVOIR DE LA PERFORMANCE AVEC MOINS DE CONSOMMATION

La diversité des CDE

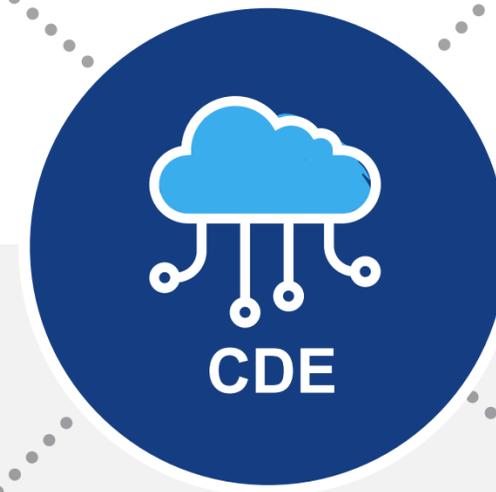
- Un éditeur
- Le code du projet
- Un conteneur d'exécution



Gitpod



GitHub
Codespaces



Jetbrains
Space



Amazon Dev
environments

Python

- Facile à apprendre
- Complet pour votre futur
- Langage de prédilection du monde de l'IA aujourd'hui ...



LangChain
pour Java

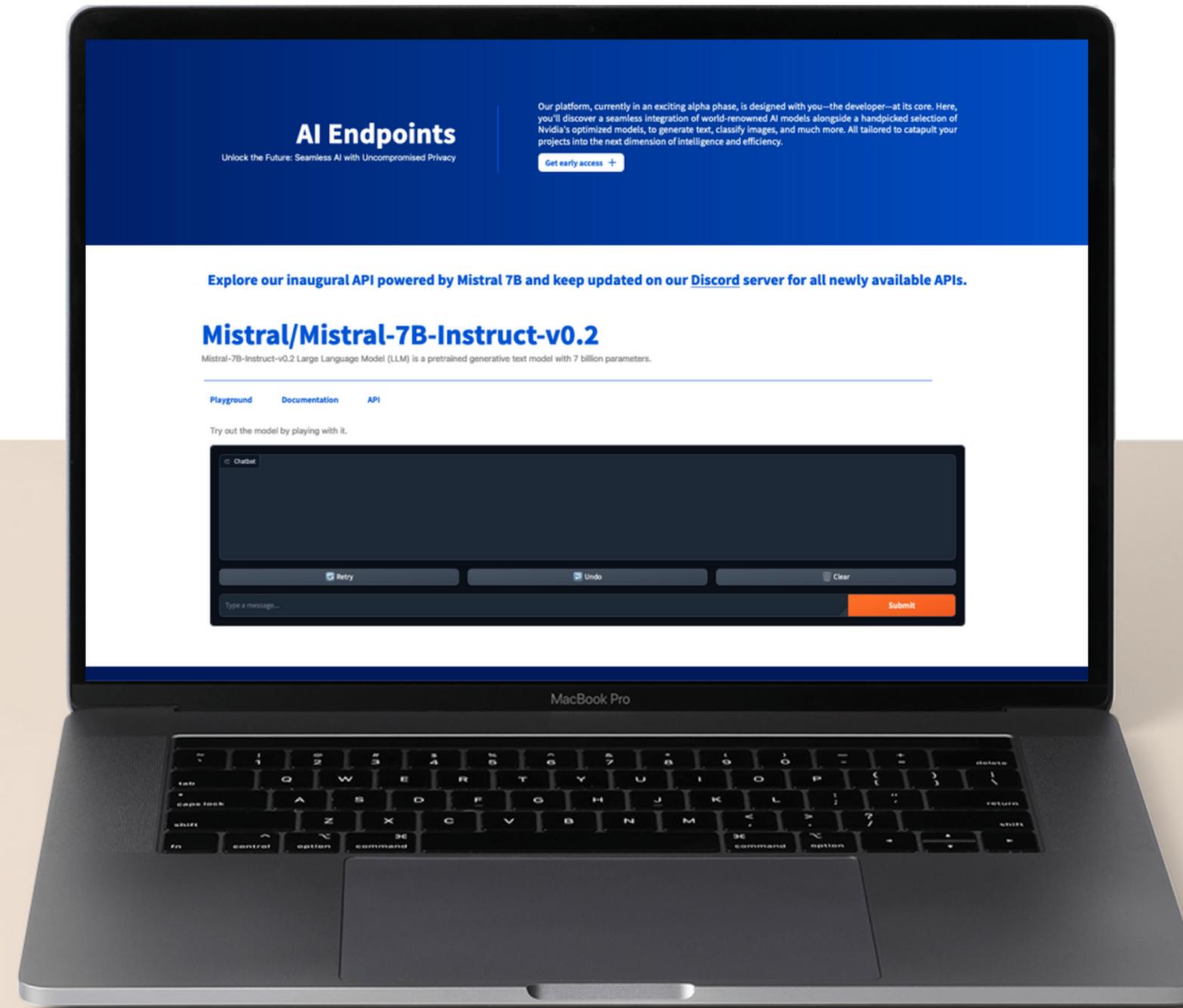


Tensorflow js

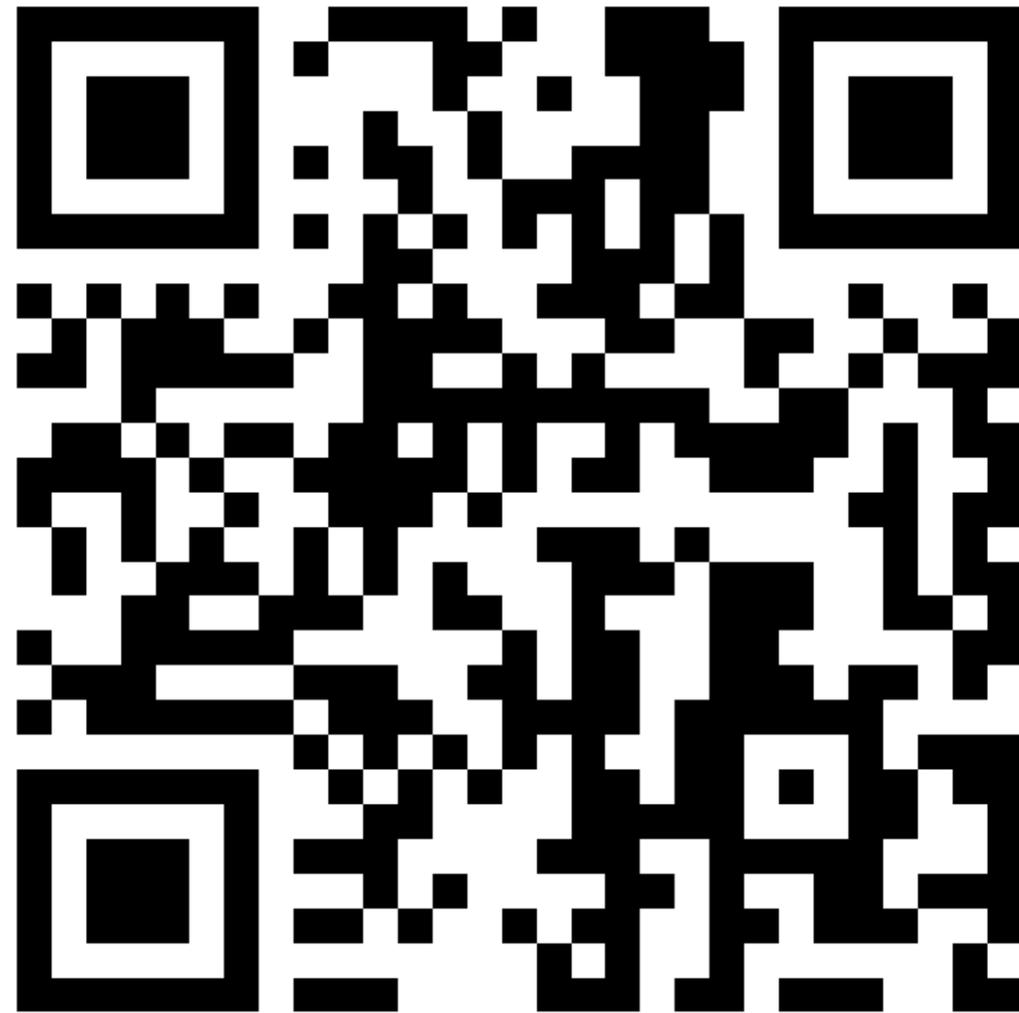


Golearn

AI Endpoints



AI Endpoints



<https://endpoints.ai.cloud.ovh.net>



Merci !!!!!



Slides

<https://ovh.to/ECDeE3h>



Feedbacks

<https://ovh.to/G6fxRg>



Code source <https://ovh.to/ofPmni6>

Ressources

- [Documentations](#) OVHcloud univers IA
- [Repository GitHub](#) du workshop
- [Repository d'exemples IA](#)
- <https://unsplash.com/>
- <https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network>
- <https://medium.com/mlearning-ai/introduction-to-neural-networks-weights-biases-and-activation-270ebf2545aa>
- <https://www.youtube.com/@MachineLearnia>
- <https://gitpod.io>

