

Viens dompter ta première IA en Python

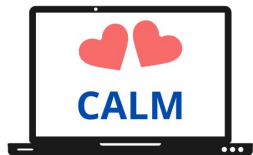


{RIVIERADEV}



MERCI !!!

Thierry Chantier



 GDG Cloud Lyon



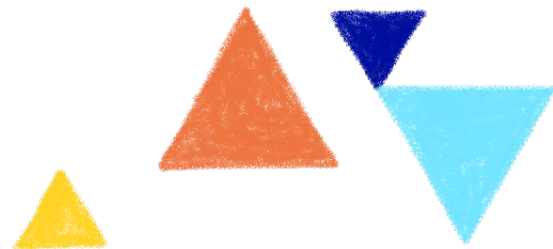
DevRel @OVHcloud



 TitiMoby@mamot.fr

 TitiMoby

 <https://noti.st/titimoby>



Stéphane Philippart



🏷️ 🥑 DeveloperAdvocate@OVHcloud 🦄

🏷️ Co-créateur de [TADx](#) (meetup à Tours)

```
➤ whoami
stef
```

🧠 Padawan Intelligence Artificielle 🍷

🐦 [@wildagsx](#)

🔗 <https://philippart-s.github.io/blog>

🦑 <https://github.com/philippart-s/>

💬 <https://www.linkedin.com/in/philippartstephane/>

TAD 






Que va-t-on voir aujourd'hui ?

 Les principes dans l'intelligence artificielle

 Le kit de survie Python pour suivre ce deep dive

 Un CDE c'est quoi et ça sert à quoi ?

 Les ressources utilisées chez OVHcloud

 En avant pour le développement : un notebook, un job d'entraînement et une application utilisant le modèle



Qui êtes vous ?

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

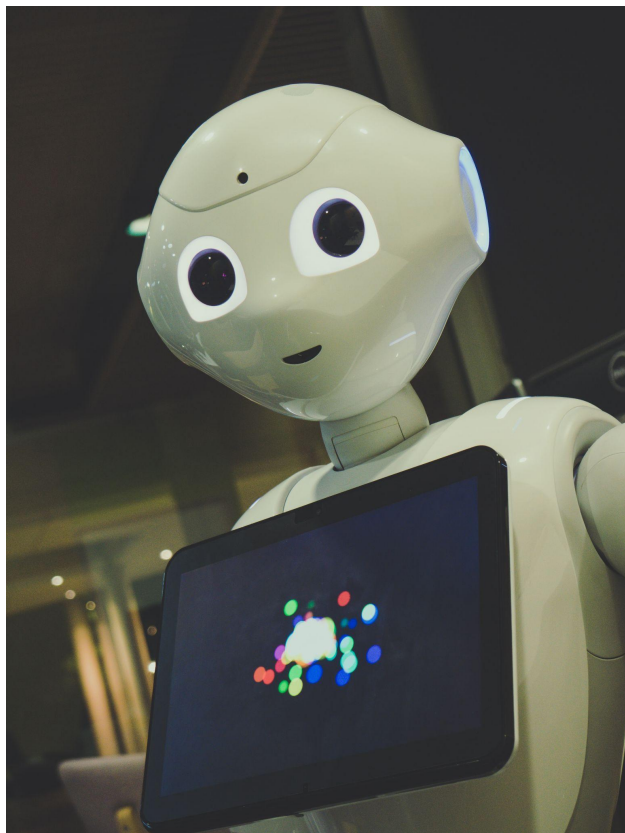




Pense bête et liens 

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





Intelligence Artificielle

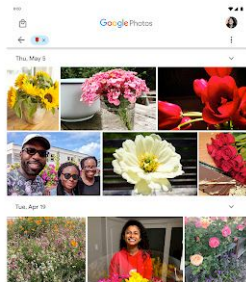
L'IA dans notre quotidien



Recommandations personnalisées de contenu



Tags automatique, recherche de contenus



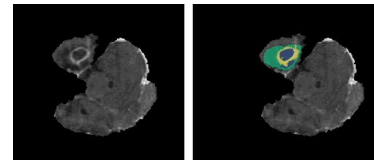
Conduite assistée et autonome



Assistant vocal

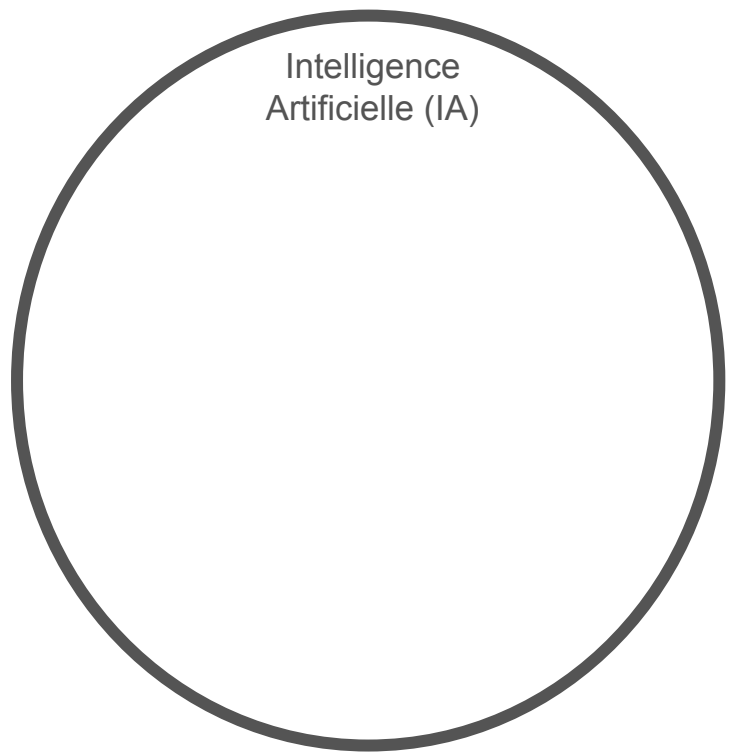


Aide aux diagnostics médicaux





Les champs de l'IA

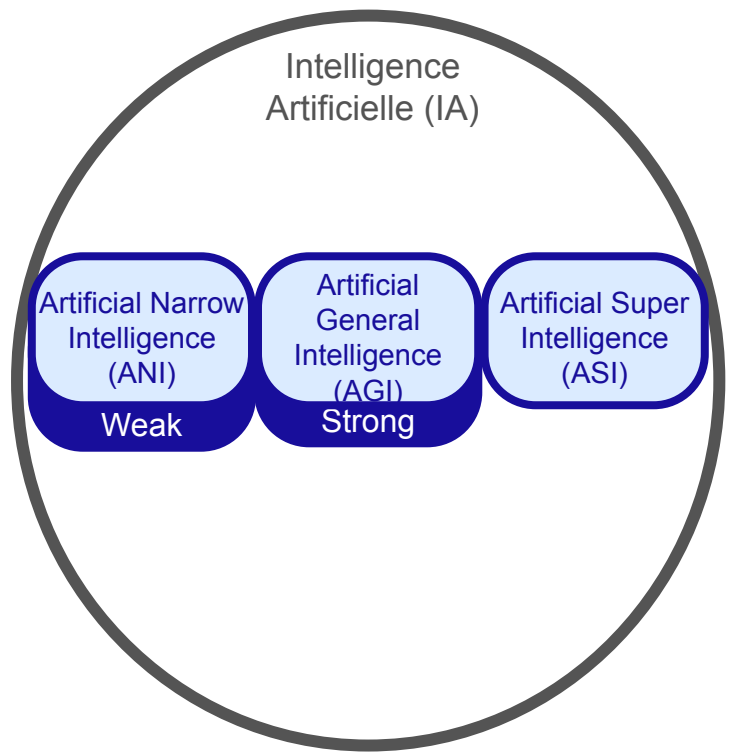


👉 IA : Techniques pour simuler les capacités “intelligentes” d’un être humain

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



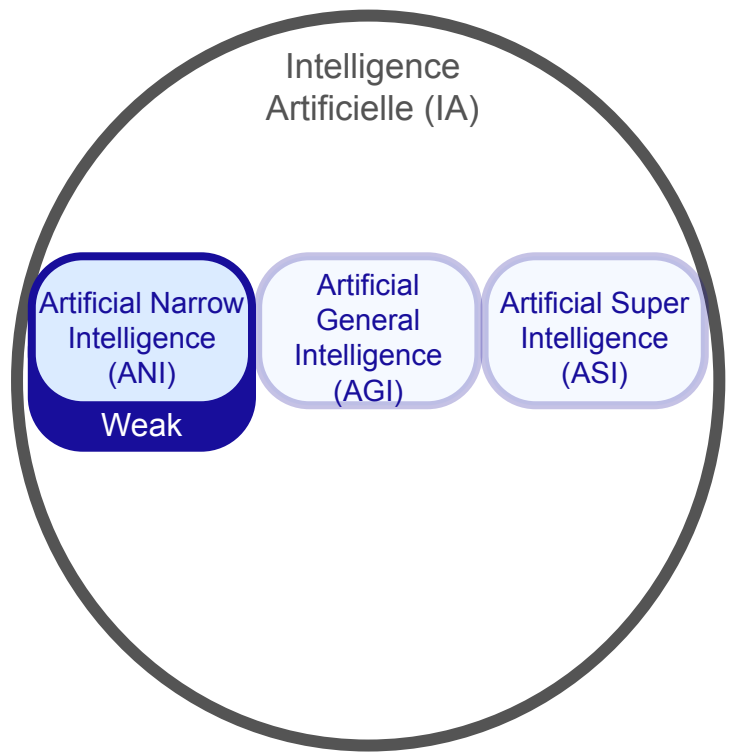
Les champs de l'IA



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



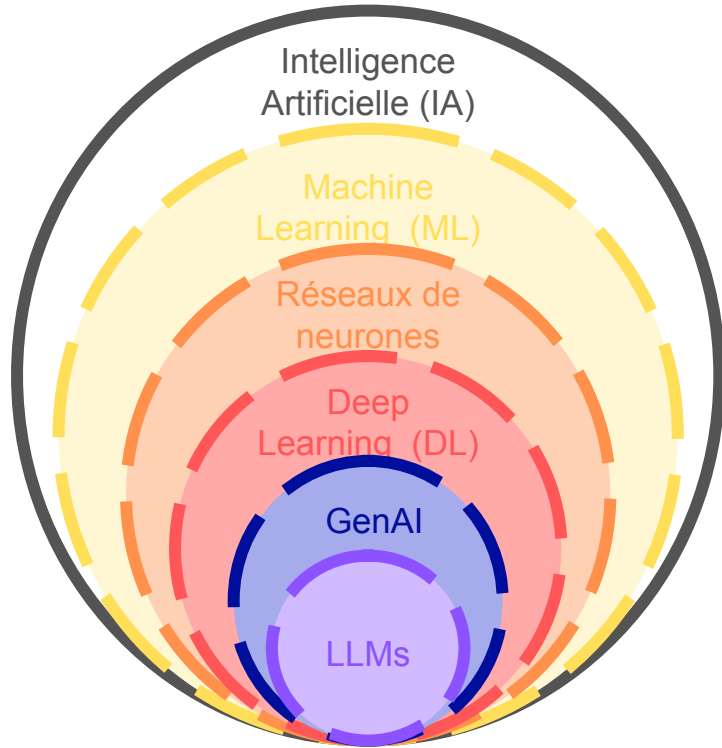
Les champs de l'IA



👉 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.

👉 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



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 transformer en Dataset (en gros une base de données pour IA) et les labelliser
-  Un dataset contient les données d'apprentissage, de validations et de tests
-  Enfin, ce sont des vecteurs et autres matrices qui seront manipulés par le modèle



Exemple d'un dataset

Label correspondant :



1 **0.650401** **0.611884** **0.694224** **0.75**

Classe personne

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

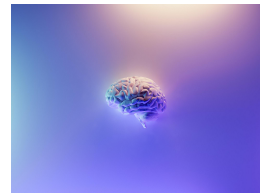


Les modèles dans l'IA

 C'est le cerveau de votre application

 C'est ici que l'on retrouve les formules mathématiques

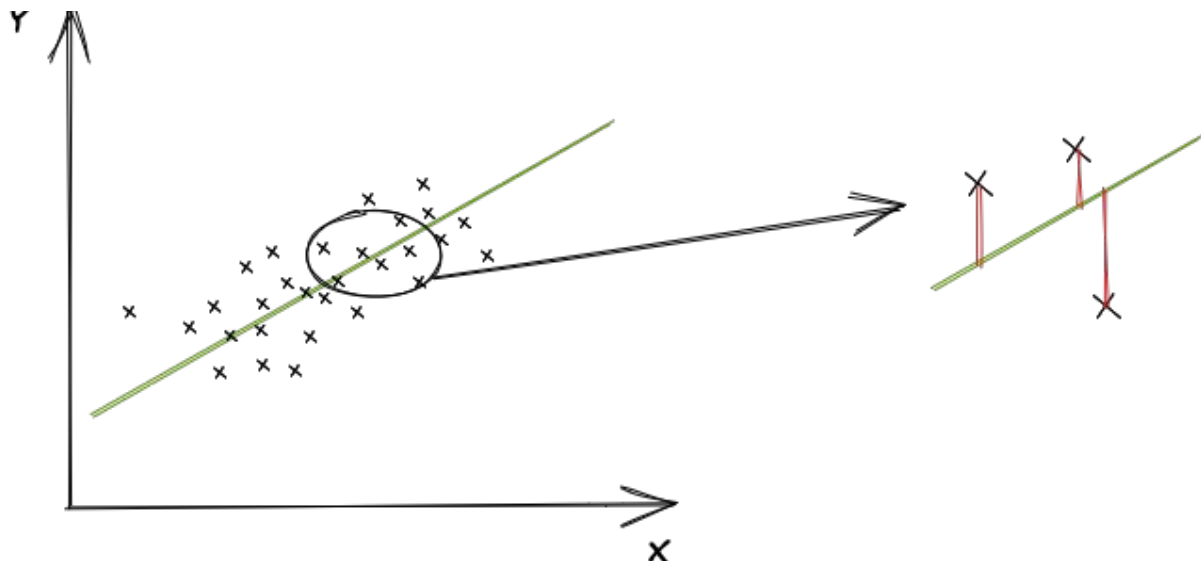
 Différents en fonction des tâches de machine learning



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



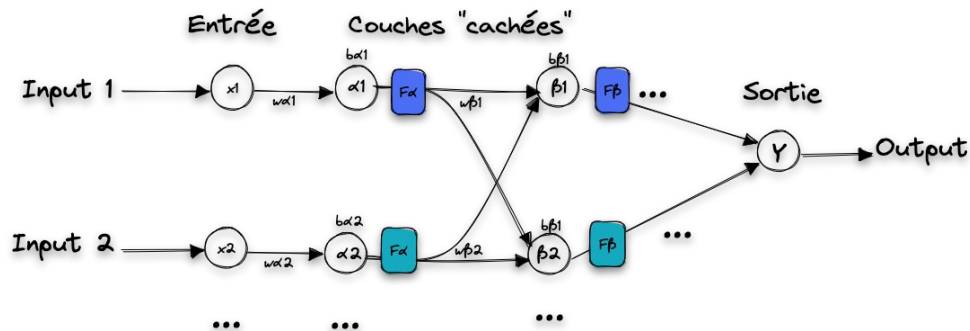


Deep learning & Réseau de neurones

👉 L'étape d'après ... Plus puissante, plus complexe

🧠 Analogie avec les neurones humain plutôt fausse

↔ Essentiellement due au mécanisme d'inter-connexions



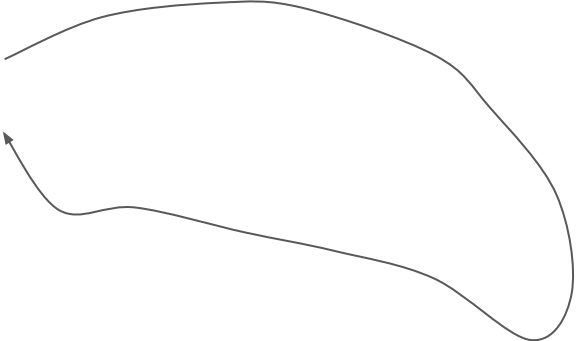
⚖️ w^* sont les poids et permettent d'ajuster le comportement du réseau

🐛 b^* sont les biais pour rendre le modèle "plus réaliste" et influencer sur la fonction d'activation

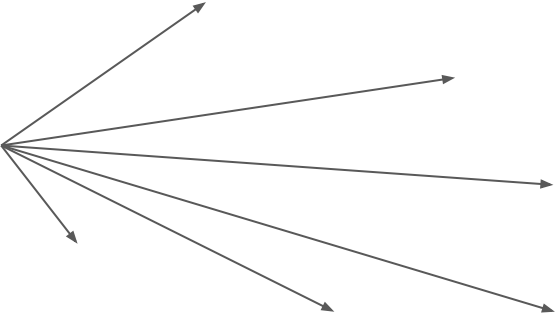
🔗 F^* sont les fonctions d'activation



GPU vs CPU

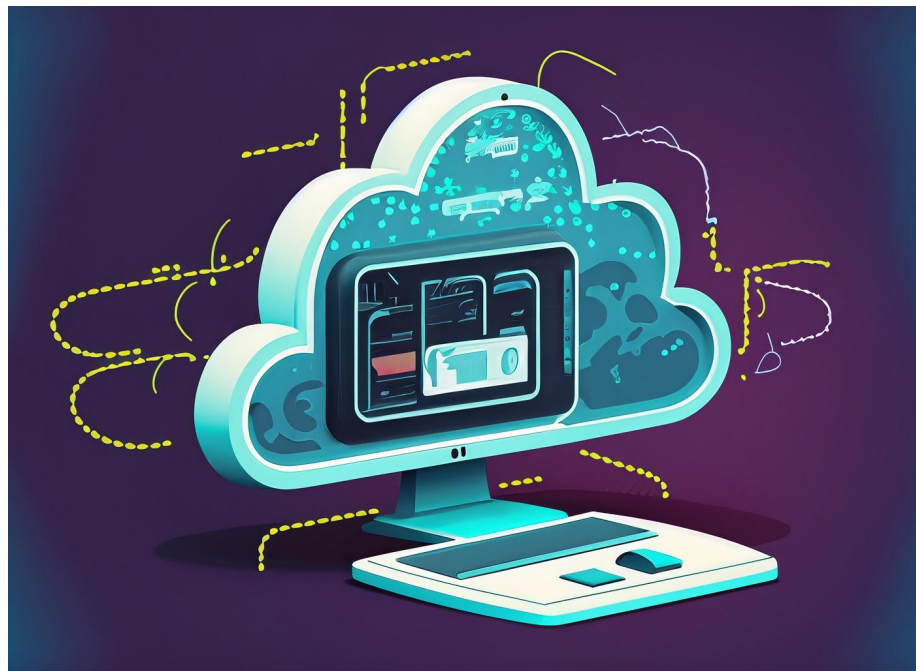


CPU



GPU

Cloud Development Environment



Cloud Development Environment



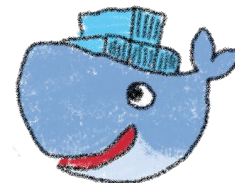
Le code du projet



Un éditeur de code



Un environnement d'exécution





A 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

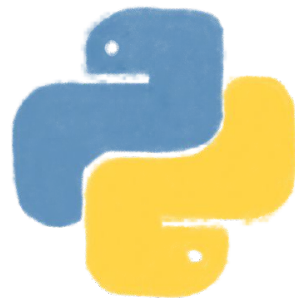


Un exemple de CDE : Gitpod

- 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
-  “All batteries included”
-  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 à OVHcloud



Public Cloud



AI Notebooks : JupyterLab et VSCode, images pré-construites



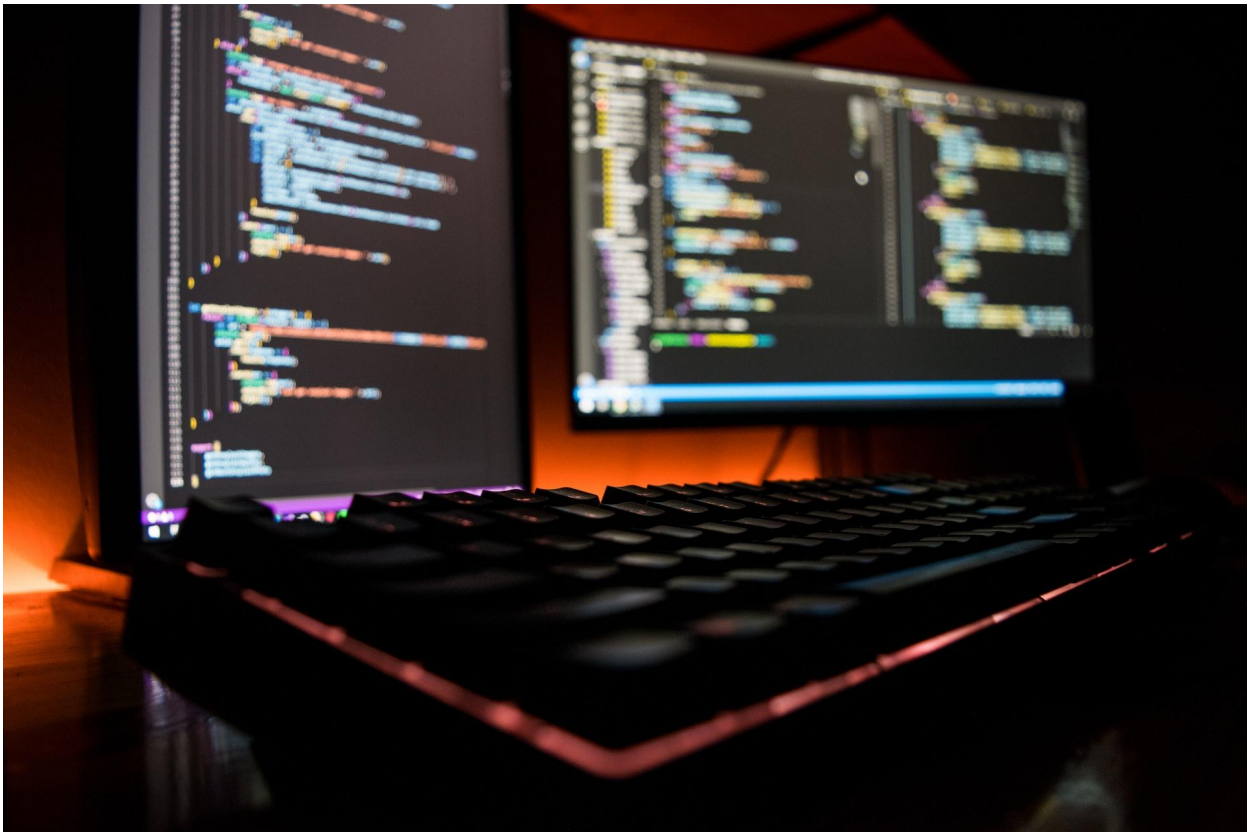
AI Training : GPU as a Service



AI Deploy : CaaS pour l'IA



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

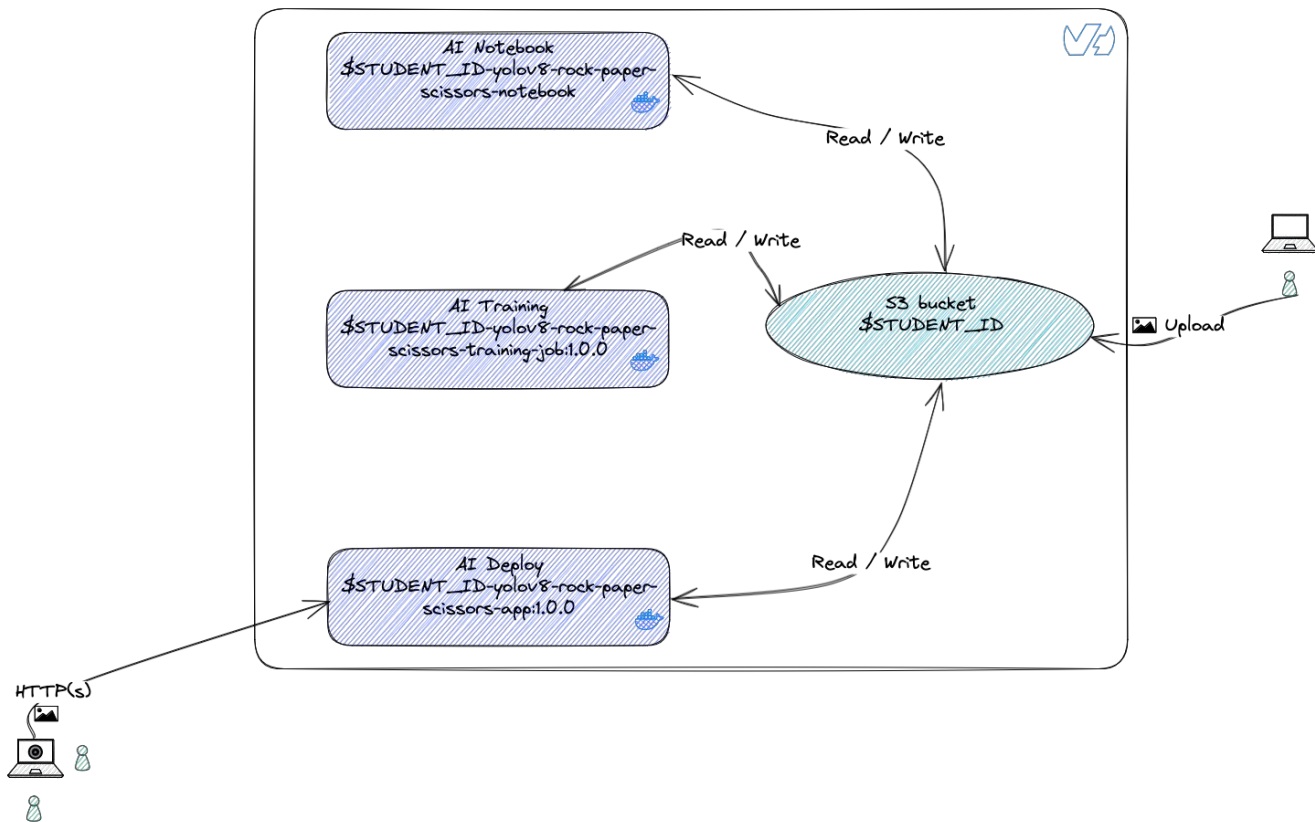


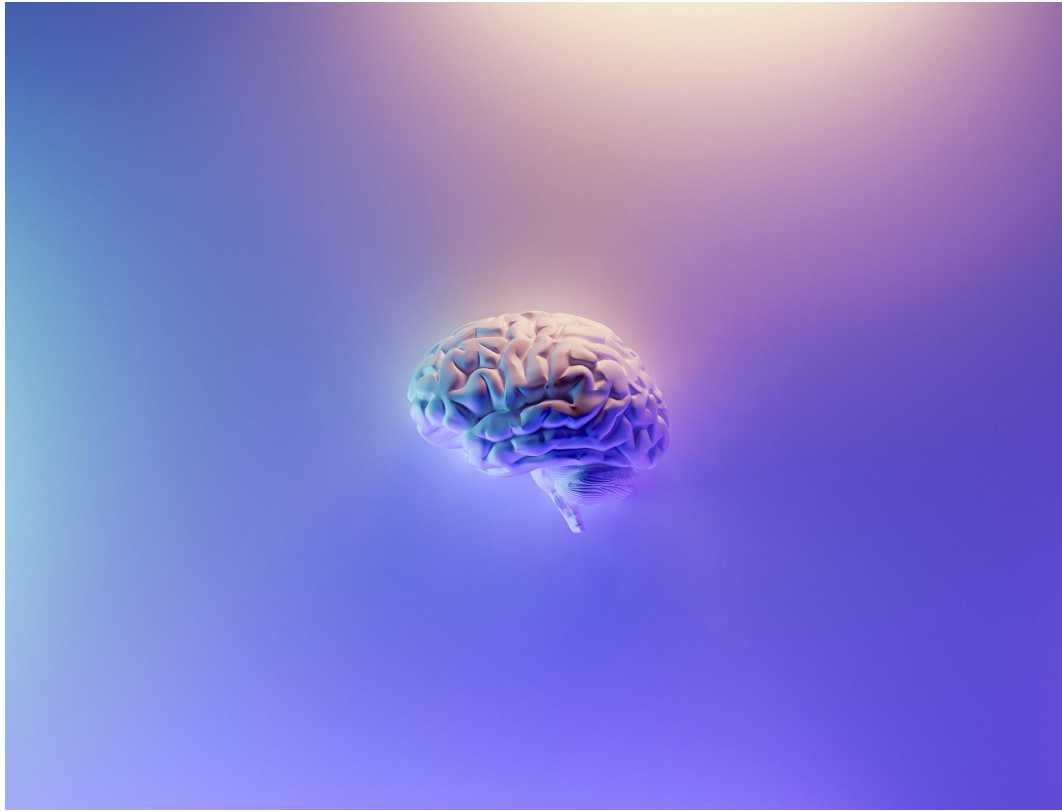
⚡ Initialisation de l'environnement GitPod

- 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 Notebook



Instructions

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

<https://ovh.to/Zz5AnB>





Et dans la console d'admin OVHcloud ?

Dashboard Bare Metal Cloud Hosted Private Cloud **Public Cloud** Web Cloud Telecom Sunrise Marketplace English 4 Stéphane Philippart

Gateway

Containers & Orchestration

- Managed Kubernetes Service
- Kubernetes Load Balancer
- Managed Private Registry
- Workflow Management

AI & Machine Learning

- AI Dashboard New
- AI Notebooks**
- AI Training
- AI Deploy New

Data & Analytics

WORKSHOP_101_AI / AI Notebooks

AI Notebooks Guides

To use AI Tools, please ensure that you are using a configured AI user or a token.
Manage my AI users and tokens

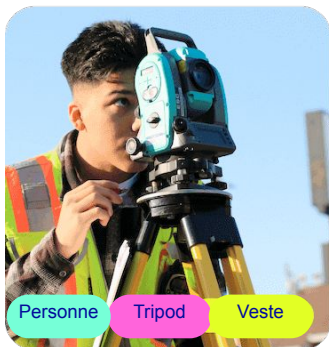
[+ Create a notebook](#) Filter

Name	Region	Environment	CPU	GPU	Privacy	Operating time	User	Status	Access
attendee-backup-ai-lab-notebook	Gravelines	Miniconda- conda-py39-cuda11.8-v22-4	13	1	Private	2mn	user-UtR7ujyZfCgm	Stopped	JupyterLab
attendee-test1-yolov8-rock-paper-scissors-notebook	Gravelines	Miniconda- conda-py39-cuda11.8-v22-4	13	1	Private	2h 30mn	user-jcP2qp858vHD	In service	JupyterLab

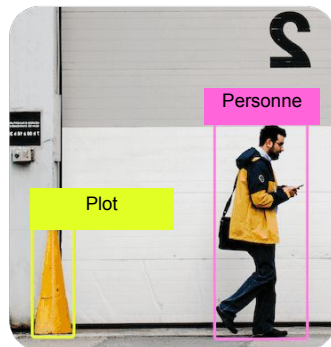


YOLOv8 (You only look once)

Classification



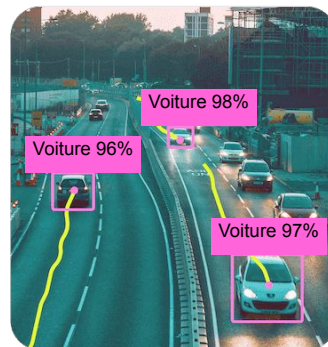
Détection



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

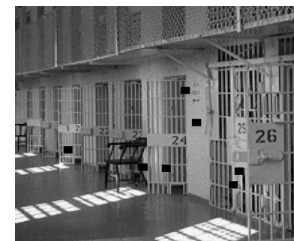
Exemple : détection d'objets divers qui devient une détection de signes





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 *attendee-conf.json* dans le workspace de projet Gitpod

💻 Via la CLI :

```
$ 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

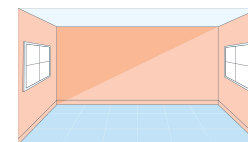
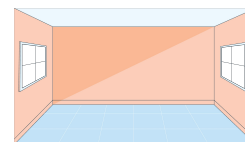
<https://xx-xx-xx-xx-xx.notebook.bhs.ai.cloud.ovh.net>



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

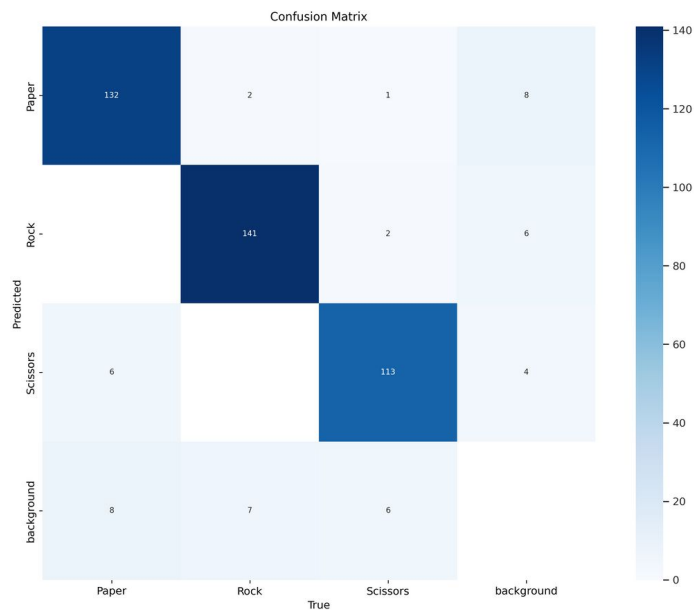
Prédiction



Matrice de confusion

Vue résumée des classifications (Nombre de TP, TN, FP, FN)

La diagonale représente les bonnes prédictions, tandis que les autres sont des erreurs.





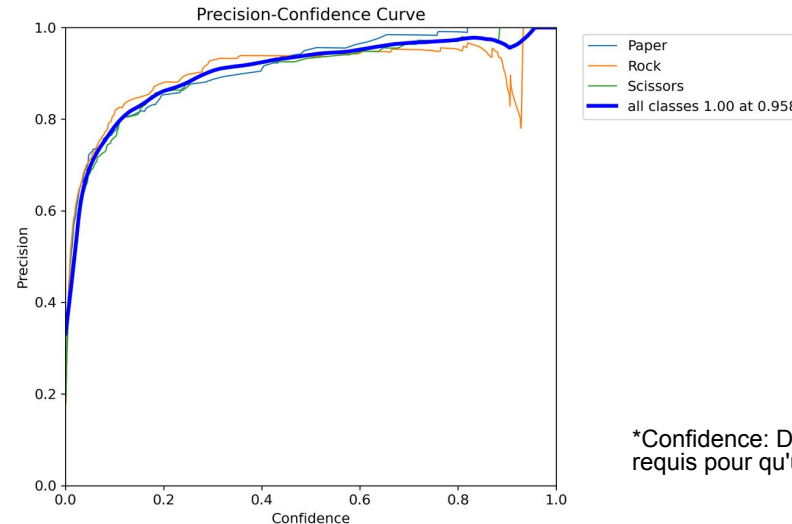
Précision

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

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



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

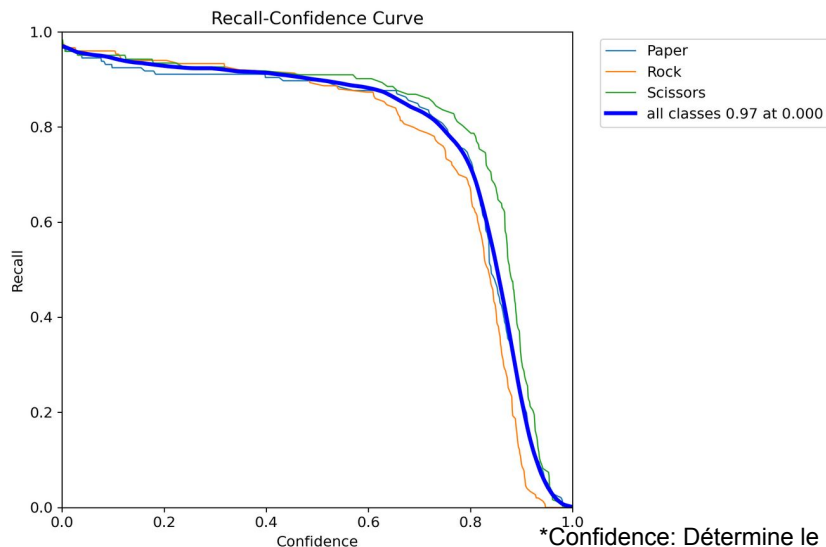


Rappel (Recall)

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

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

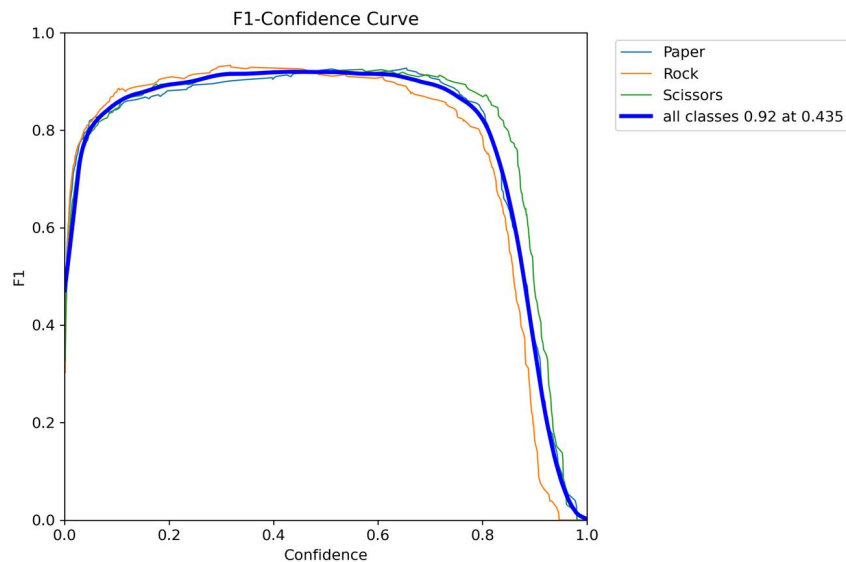


*Confidence: Détermine le score de confiance minimum requis pour qu'une prédiction soit considérée comme valide



Score F1

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



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



Bravo on a créé notre premier modèle
d'intelligence artificielle !!





Entraînement avec AI Training



Instructions

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

<https://ovh.to/tFHguV>



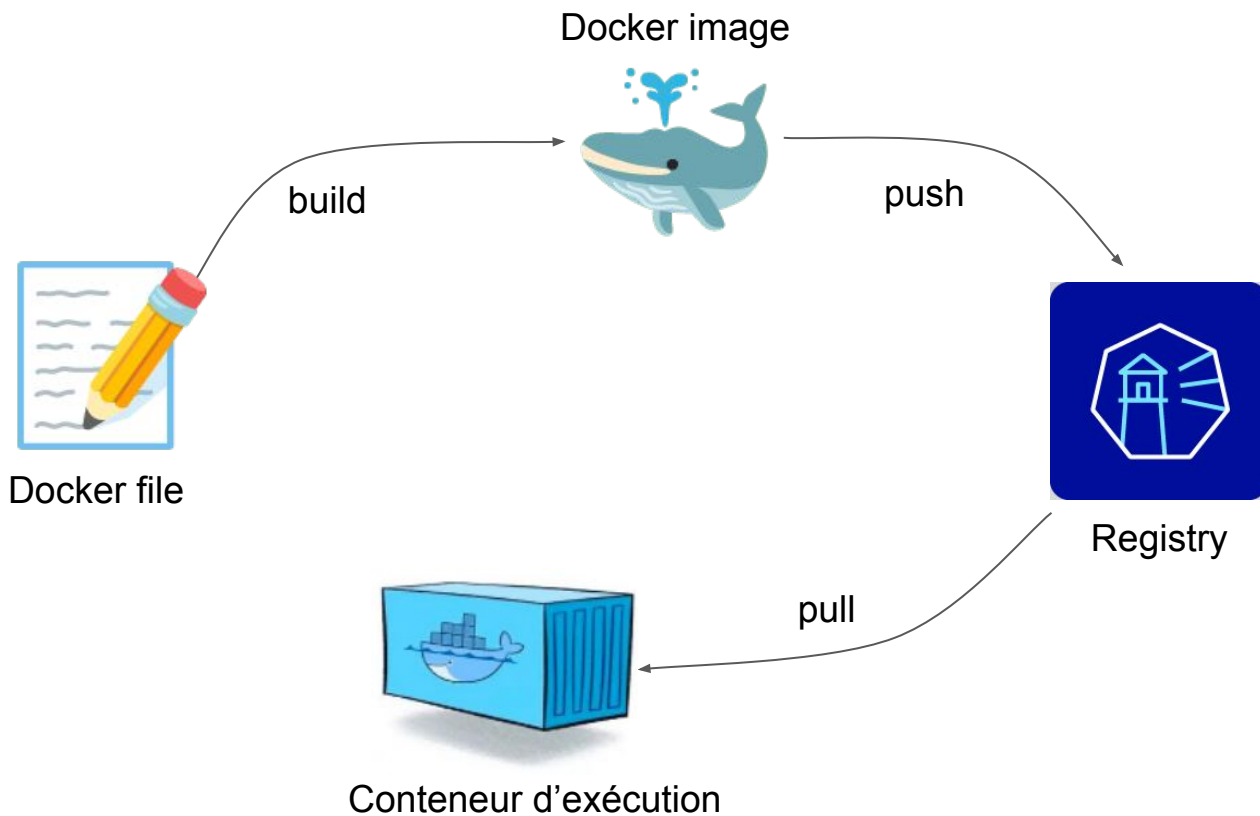
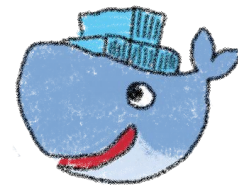


Rappels

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



Docker





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



⚡ Création du Job

💻 Avec la CLI :

```
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/attendee:RW:cache \  
  --unsecure-http \  
  wilda/yolov8-rock-paper-scissors-training-job:1.0.0
```



Et dans la console d'admin OVHcloud ?

The screenshot shows the OVHcloud Admin Console interface. The top navigation bar includes 'Dashboard', 'Bare Metal Cloud', 'Hosted Private Cloud', 'Public Cloud', 'Web Cloud', 'Telecom', 'Sunrise', and 'Marketplace'. The user is logged in as 'Stéphane Philippar'. The left sidebar shows the 'AI & Machine Learning' section, with 'AI Training' highlighted. The main content area is titled 'AI Training' and contains a message: 'To use AI Tools, please ensure that you are using a configured AI user or a token. Manage my AI users and tokens'. Below this, there is a command line instruction: 'To access this list of jobs via the command line, run the following command: `ovhai job ls`'. A 'Launch a new job' button is visible. A table lists the AI training jobs:

ID	Name	Region	Docker Image	Use
5a8e3bd7-65d5-4b9e-8c95-a91fb39e016c	attendee-0-yolov8-rock-paper-scissors-training-job	GRA	ubf2r2if.c1.gra9.container-registry.ovh.net/lab/0/yolov8-rock-paper-scissors-training-job:1.0.0	use



Plan B

```
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/attendee:RW:cache \  
  --unsecure-http \  
  wilda/yolov8-rock-paper-scissors-training-job:1.0.0
```



Suivi des logs

```
ovhai job logs -f <job id> --token $AI_TOKEN
```



Bravo on a entraîné notre premier modèle
d'intelligence artificielle !!





Welcome on the 🗑️ 📄 ✂️ game!

Take your picture in real time:



Take Photo

Application avec AI Deploy



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 :

```
ovhai app run \  
  --token $AI_TOKEN \  
  --name $STUDENT_ID-yo!ov8-rock-paper-scissors-app \  
  --cpu 1 \  
  --default-http-port 8501 \  
  --volume $STUDENT_ID@S3GRA:/workspace/attende:RW:cache \  
  --unsecure-http \  
  $REGISTRY_NAME/$STUDENT_ID/yo!ov8-rock-paper-scissors-app:1.0.0
```




Et dans la console d'admin OVHcloud ?



Dashboard Bare Metal Cloud Hosted Private Cloud **Public Cloud** Web Cloud Telecom Sunrise Marketplace

English 4 Stéphane Philippart

Gateway

Containers & Orchestration

- Managed Kubernetes Service
- [Kubernetes Load Balancer](#)
- Managed Private Registry
- Workflow Management

AI & Machine Learning

- AI Dashboard New
- AI Notebooks
- AI Training
- [AI Deploy](#) New

WORKSHOP_101_AI / AI Deploy / My apps

AI Deploy

Guides

To use AI Tools, please ensure that you are using a configured AI user or a token.
[Manage my AI users and tokens](#)

[+ Deploy an app](#)

Name	Image	Region	Resources deployed	Creation date	Last updated	Status
attendee-0-yolov8-rock-paper-scissors-app	ubf2r2if.c1.gra9.container-registry....	Gravelines	1 CPU / 1 replicas	28/09/2023 13:02	28/09/2023 13:11	Running

25 of 1 results



Plan B

```
ovh ai app run \  
  --token $AI_TOKEN \  
  --name $STUDENT_ID-yolov8-rock-paper-scissors-app \  
  --cpu 1 \  
  --default-http-port 8501 \  
  --volume $STUDENT_ID@S3GRA:/workspace/attendee:RW:cache \  
  --unsecure-http \  
  wilda/yolov8-rock-paper-scissors-app:1.0.0
```



Suivi des logs

```
ovhai app logs -f <app id> --token $AI_TOKEN
```

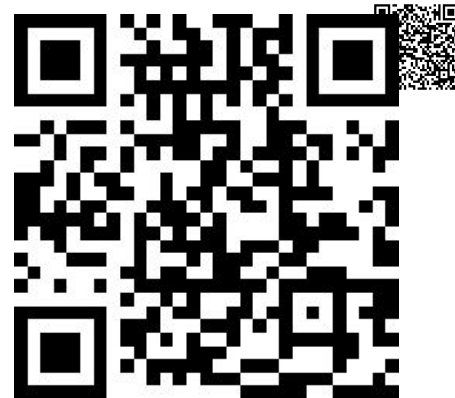
Accéder à l'application

 Avec la CLI :

```
$ ovhai app get <AppId> --token $AI_TOKEN
```

Status:

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



<https://ovh.to/fRZW8kp>



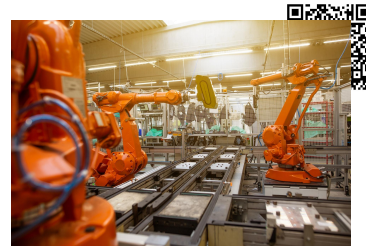
 Bravo on a créé notre première
application basée sur l'intelligence artificielle
!! 



Take away / Next ?



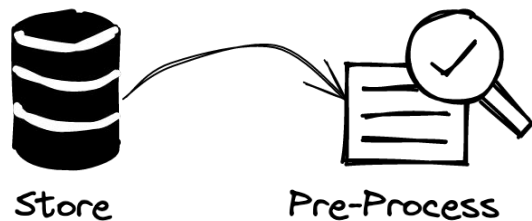
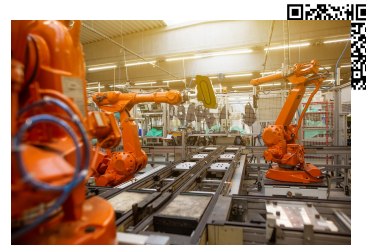
Un workflow typique dans l'IA



Store

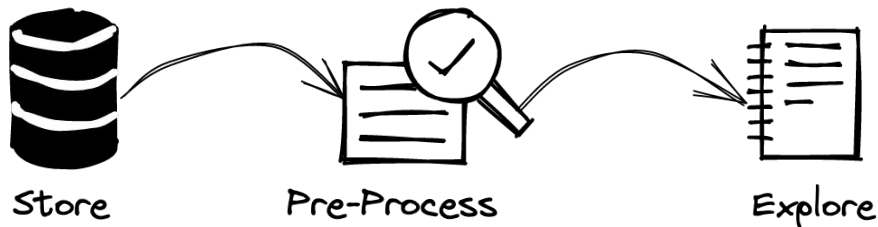
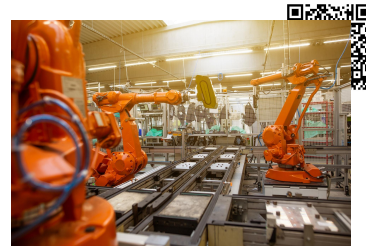
- 📀 Il faut une très grande quantité de données
- 📀 La plupart du temps la donnée est brute
- ⚠️ Attention aux coûts (stockage, lecture / écriture)

Un workflow typique dans l'IA




- 🔧 Créer des datasets nettoyés (Valeurs manquantes, Normalisation, ...)
- 🔧 Il existe des datasets pré-crés (gratuits ou payants)
- 🎯 Le but est de traduire les données brutes dans un langage compréhensible par le modèle

Un workflow typique dans l'IA




 Le “cerveau” de l’application

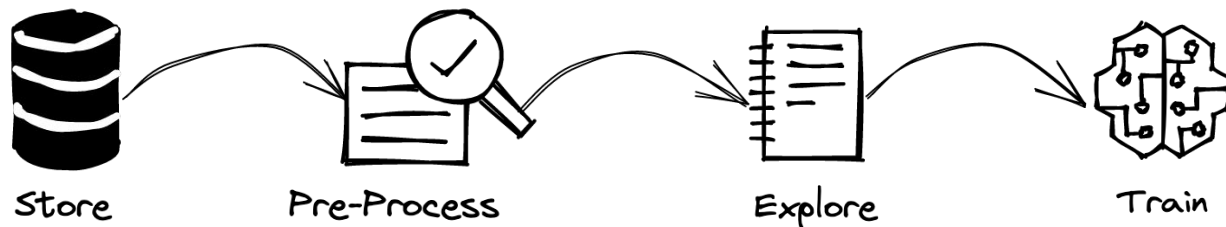
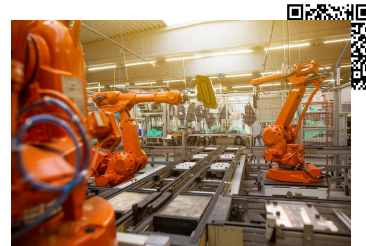
 Basé sur des opérations mathématiques complexes

 Réseaux de neurones artificiels

 Fait par les Data Scientist

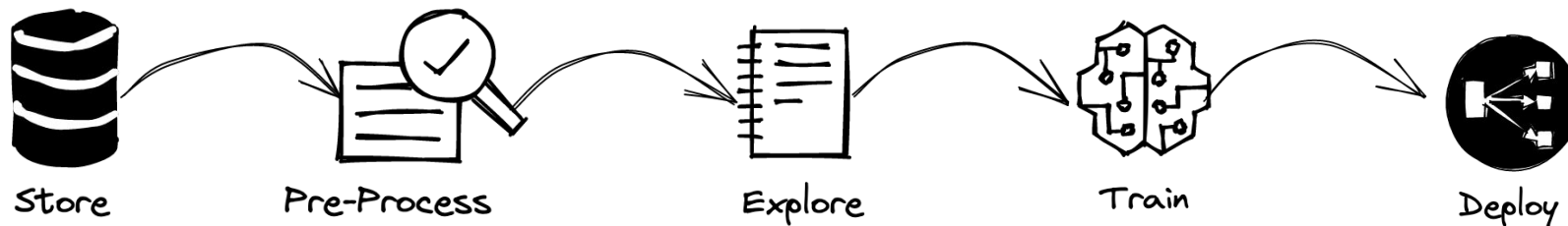
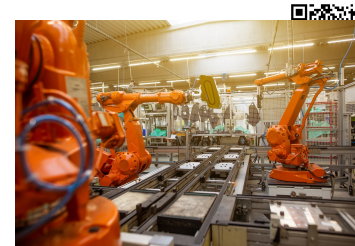
 Utilise des éditeurs (JupyterLab, Matlab, VSCode, ...)

Un workflow typique dans l'IA



- ✨ 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 dans l'IA



 Utilise le modèle pour faire l'inférence

 Peut exposer une API ou une interface utilisateur



AI Endpoints en quelques mots

 LLM as a Service

 Alpha ~ juin 2024 (gratuit)

 Dispo des LLM via des API

 Catégories des modèles

- Assistants : CodeLlama 13b, Llama 3 70b, Mixtral 8x22b, ...
- Embedding : BGE base, Multilingual E5, ...
- NLP: Bart, Bert, ...
- Translation : T5 large, ...
- Audio : Nvidia RIVA ASR
- Vision : Yolo V8, ...

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



LangChain ... et ses variantes



Le projet originel : [LangChain](#)

- Technologies : Python et Javascript
- SDK / Librairie
- Faciliter l'adoption et l'utilisation des LLM
- Accès par API aux modèles
- Templating de prompt
- RAG
-



Les projets dérivés : [LangChain4j](#), [LangChainGo](#), ...

- Aucun lien avec la communauté LangChain existante
- Base commune mais qui (peut) diverger du LangChain d'origine
- Mêmes objectifs de simplification que le LangChain d'origine



Les métiers

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



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



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

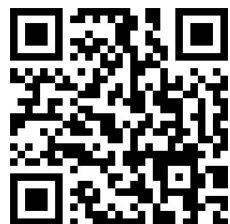




Python



- Facile à apprendre
- Complet pour votre futur
- Langage de prédilection du monde de l'IA aujourd'hui ...
 - ... d'autres langages sont prêts pour demain
 - Chaque contexte mérite son langage
 - Pas de langage miracle universel



Langchain pour Java



Tensorflow js



Golearn



Merci !!!!



Slides

<https://ovh.to/Zz5mfBB>

Feedbacks

<https://ovh.to/FEMVsTz>



Code source

<https://ovh.to/ofPmni6>

Time to play !

<https://ovh.to/G6fxCNB>



Ressources (liens, ...)

Liens

[🔗 Documentations](#) OVHcloud univers IA

[🔗 Repository GitHub](#) du workshop

[🔗 https://unsplash.com/](https://unsplash.com/)

[🔗 https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network](https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network)

[🔗 https://medium.com/mllearning-ai/introduction-to-neural-networks-weights-biases-and-activation-270ebf2545aa](https://medium.com/mllearning-ai/introduction-to-neural-networks-weights-biases-and-activation-270ebf2545aa)

[🎥 https://www.youtube.com/@MachineLearnia](https://www.youtube.com/@MachineLearnia)

[🔗 https://gitpod.io](https://gitpod.io)

