

# Viens dompter ta première IA en Python



@titimoby



@Wildagsx

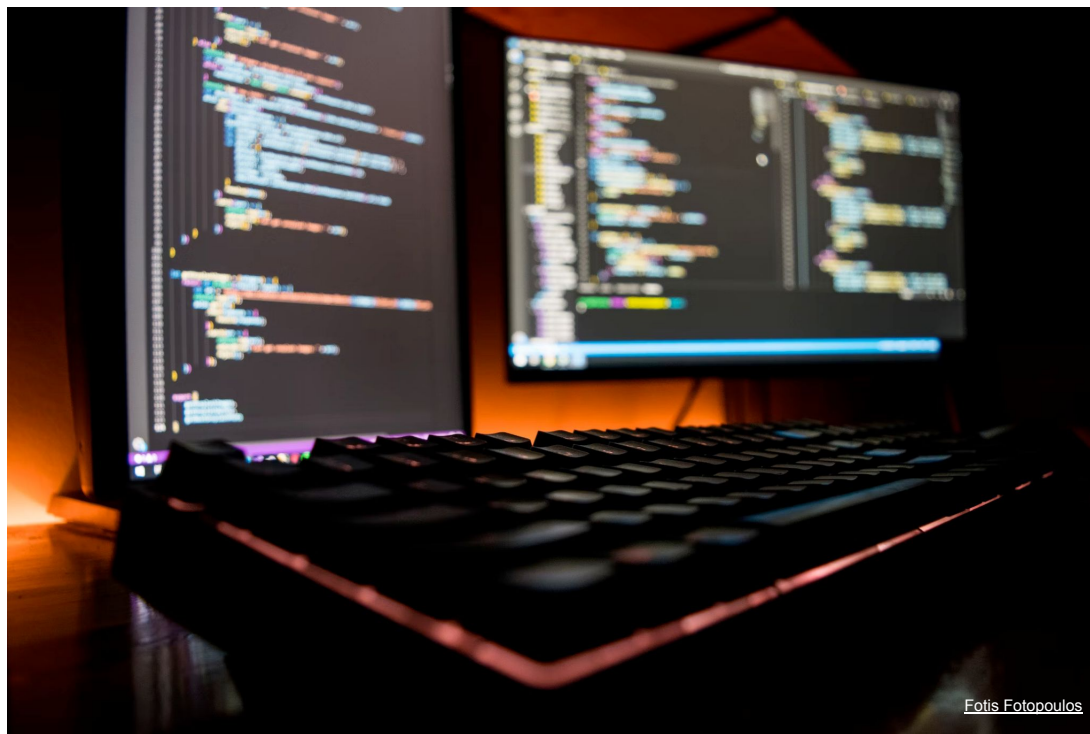


Pense bête et liens 

<https://ovh.to/Zz5AnhE>



# Combien de personnes participent au lab ?

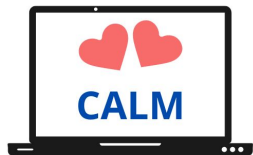


Fotis Fotopoulos



MERCI !!!

# Thierry Chantier



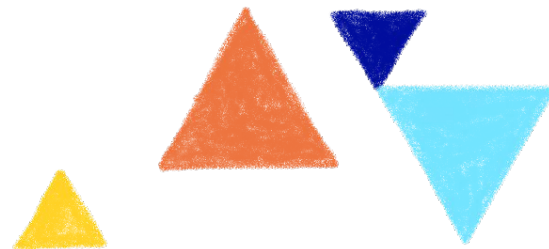
DevRel @OVHcloud



 TitiMoby@mamot.fr

 TitiMoby

 <https://noti.st/titimoby>



# Stéphane Philippart

🛒 🥑 DeveloperRelations@OVHcloud 🦄

🛒 Co-creator of [TADx](#)

(Agile, Dev, DevOps meetups in 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 workshop

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

 Les ressources utilisées chez OVHcloud

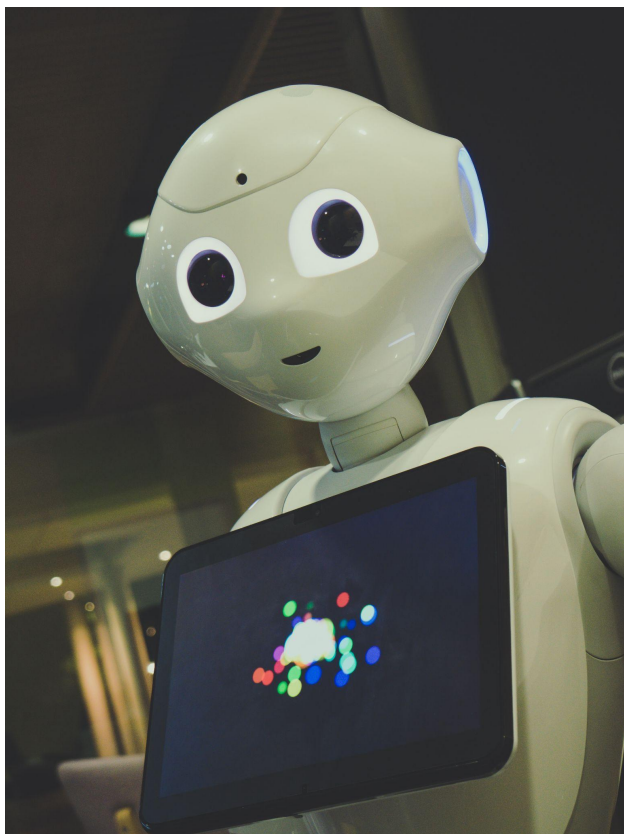
 En avant pour le développement : un notebook, un job d'entraînement et une application utilisant le modèle

# Qui êtes vous ?

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





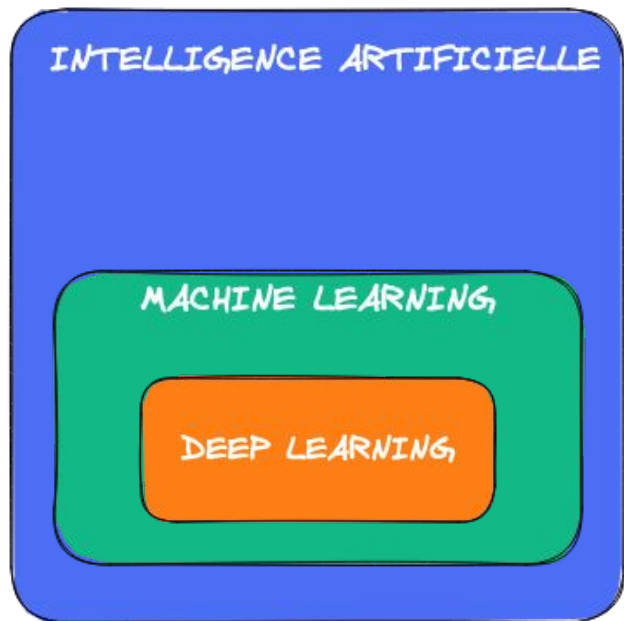


# Intelligence Artificielle

# Les différentes catégories de l'IA

- 👉 Actuellement IA ~ Machine Learning
- 👉 Artificial Narrow Intelligence (ANI) / Weak AI
- 👉 Artificial General Intelligence (AGI) / Strong AI
- 👉 Artificial Super Intelligence (ASI)

# Apprendre ...

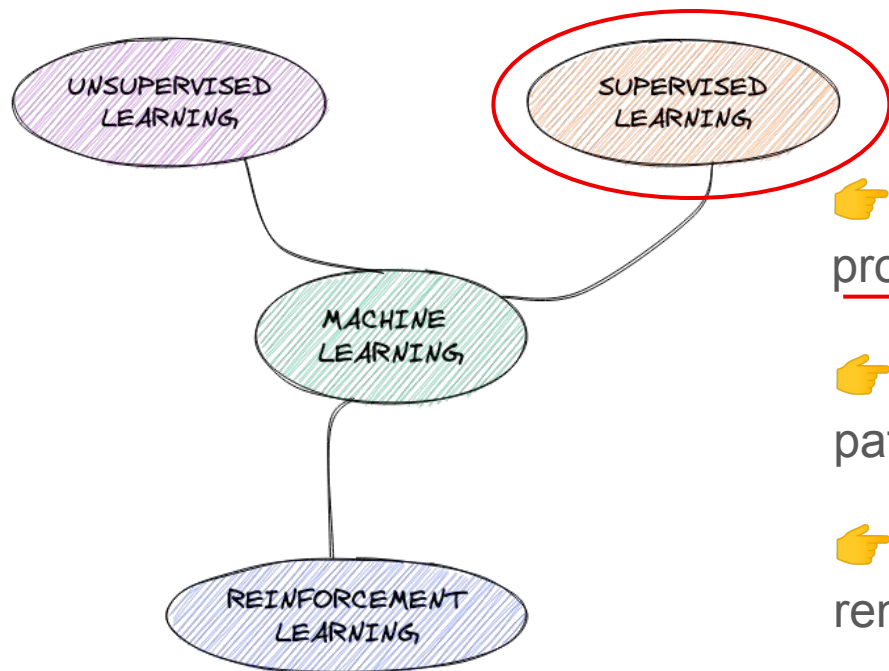


👉 IA : Techniques pour simuler les capacités “intelligentes” d’un être humain

👉 ML : Techniques pour qu’une machine apprenne sans algorithme supplémentaire par l’ingestion de données d’exemples

👉 DL : ML utilisant poussée des réseaux de neurones pour améliorer les performances dans les déductions

# Le Machine Learning dans le détail



👉 Supervised learning : apprentissage proche de ce que l'on fait avec l'Homme

👉 Unsupervised learning : apprentissage par patterns et regroupements

👉 Reinforcement learning: apprentissage par renforcement avec système de “récompenses”



# 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 :

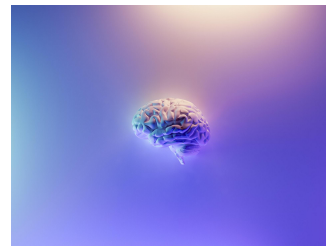


① 0.64296875 0.63046875 0.10078125 0.175

Classe personne

Coordonnées du carré

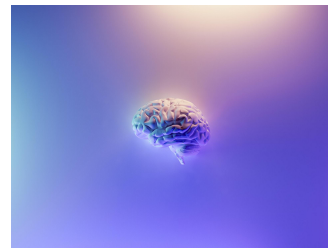
# Les modèles dans l'IA



 C'est le cerveau de votre application

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

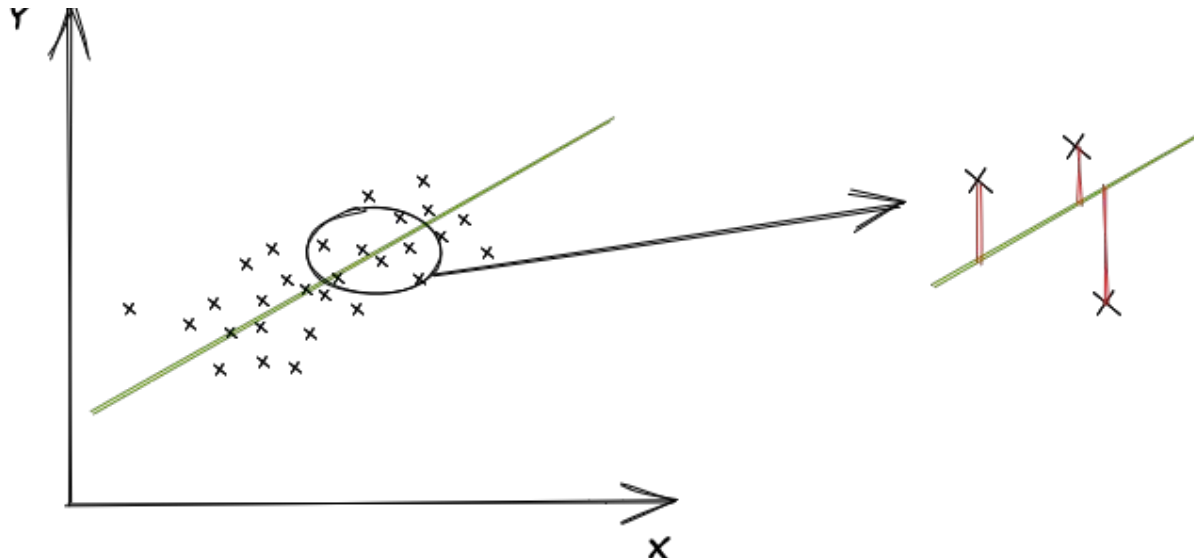
 Différents en fonction des tâches de machine learning



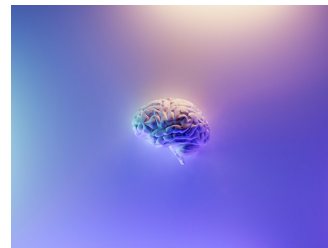
# Le coût associé au modèle

✨ 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





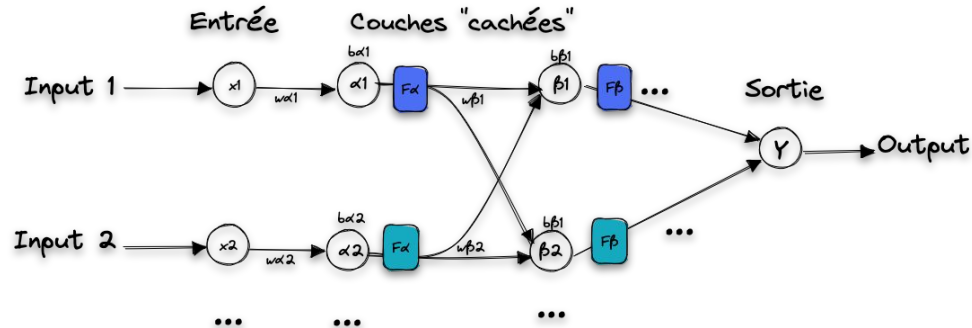


# Deep learning & Réseau de neurones

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

🧠 Analogie avec les neurones humain plutôt fausse

↔ Essentiellement due au mécanisme d'inter-connexions

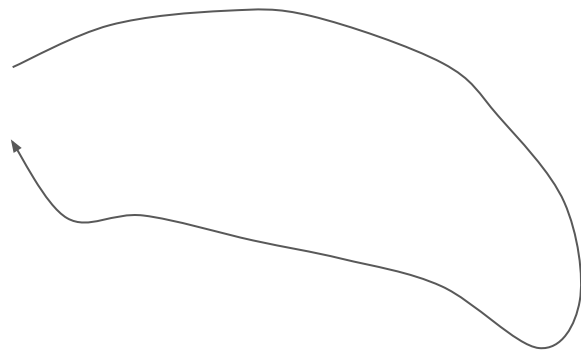


⚖️  $w^*$  sont les poids et permettent d'ajuster le comportement du réseau

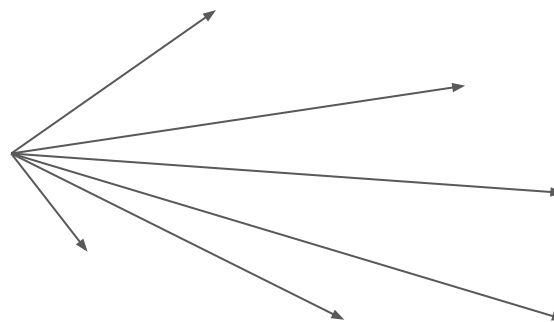
🐛  $b^*$  sont les biais pour rendre le modèle "plus réaliste" et influencer sur la fonction d'activation

🔗  $F^*$  sont les fonctions d'activation

# GPU vs CPU



CPU



GPU

# Cloud Development Environment



# Cloud Development Environment

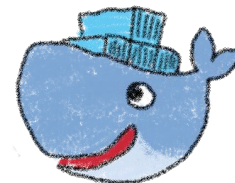
Le code du projet



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 😊)

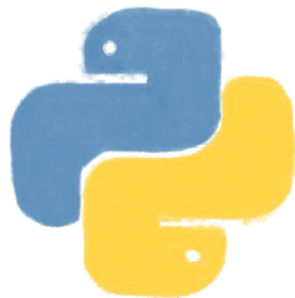
# Reproductible

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

# 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
-  “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

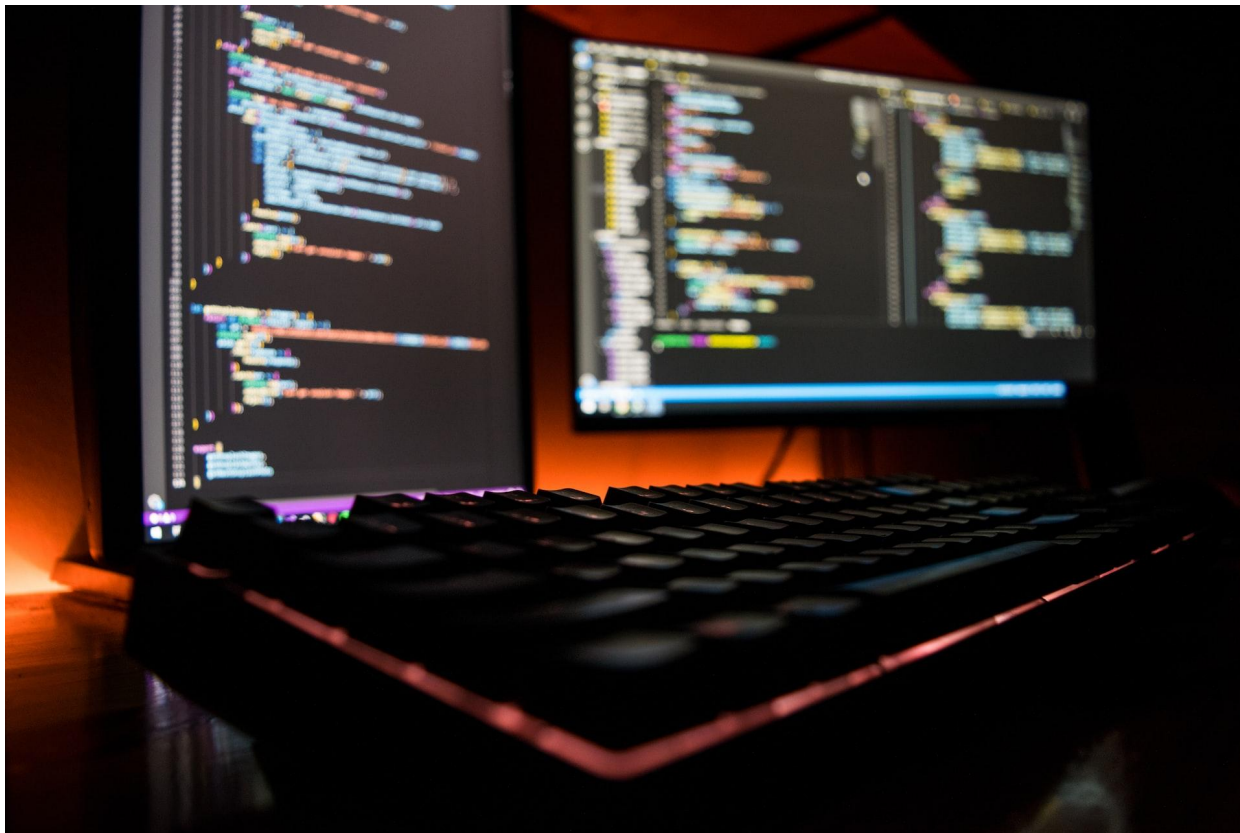


AI Training : GPU as a Service



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

# ⚙ Utiliser l'application de gestion de configuration 1/3

Rappel de l'URL : <http://162.19.64.158/q/swagger-ui/>

S'authentifier : (attendee / pass)

**PAS NÉCESSAIRE GRÂCE À GITPOD !!**



## Available authorizations

### SecurityScheme (http, Basic)

Authentication

Username:

Password:

# ⚙ Utiliser l'application de gestion de configuration 2/3

Générer la configuration

Utiliser le end-point `/lab-information` (GET)

*PAS NÉCESSAIRE GRÂCE À GITPOD !!*

**Lab Information Service**

- GET** `/lab-information` Get the next available configuration
- DELETE** `/lab-information` Clear all the configurations.
- POST** `/lab-information/users` Create a set of configurations
- PATCH** `/lab-information/users` Add a set of configurations
- GET** `/lab-information/{id}` Get a configuration given an ID.

**Lab Information Service**

**GET** `/lab-information` Get the next available configuration

**Parameters**

No parameters

**Responses**

Code	Description	Links
401	Not Authorized	No links
403	Not Allowed	No links



# ⚙️ Utiliser l'application de gestion de configuration 3/3

Cliquer sur **Execution**

Récupérer la configuration

## Lab Information Service

GET /lab-information Get the next available configuration 🔒 ↗

Parameters Cancel

No parameters

**Execute**

Responses

Code	Description	Links
------	-------------	-------

*PAS NÉCESSAIRE GRÂCE À GITPOD !!*

Parameters Cancel

No parameters

**Execute** Clear

Responses

Curl

```
curl -X 'GET' \  
  'http://162.19.64.158/lab-information' \  
  -H 'accept: application/json' \  
  -H 'Authorization: Basic dWSpY29ybJpwYXNz'
```

Request URL

```
http://162.19.64.158/lab-information
```

Server response

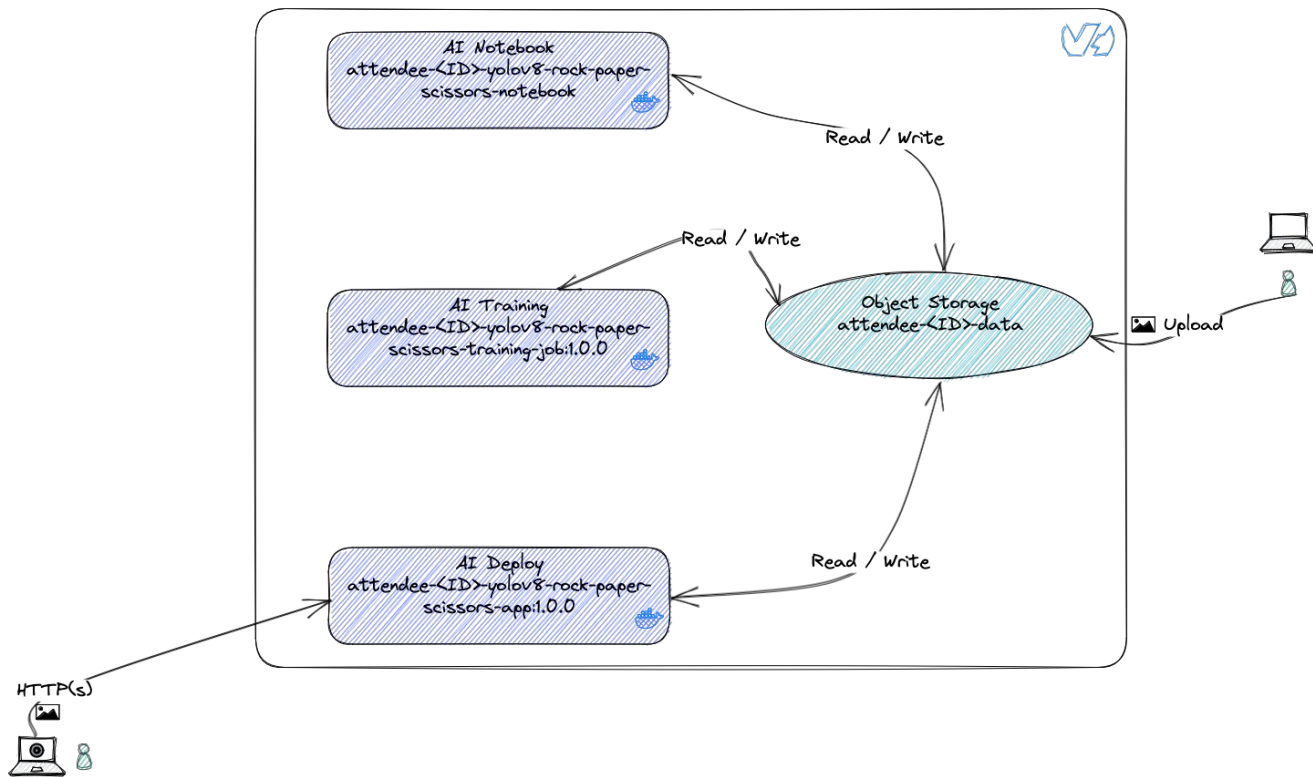
Code	Details
200	<p>Response body</p> <pre>{   "username": "xxxx",   "password": "yyyy",   "objectStorage": "attende-0-data",   "studentId": 0,   "registryLogin": "zzzz",   "registryPassword": "www",   "datasetUrl": "http://dataset.url" }</pre> <span>Download</span>

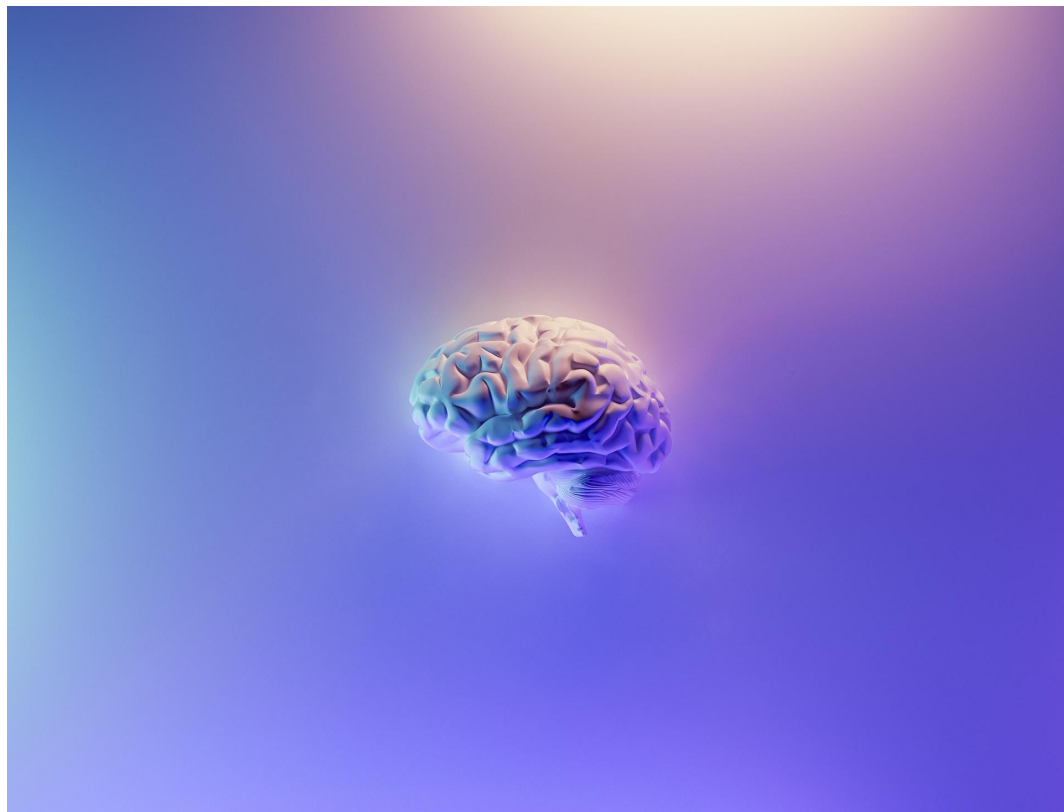


# ⚡ Initialisation de l'environnement GitPod

- Créer un compte GitPod (se connecter avec son compte GitHub)
- Prendre le modèle **large**

# Architecture de l'application





# Modèle avec AI Notebook



## Instructions

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

<https://ovh.to/MDaTnP3>



# Authentication via la CLI ovhai

ovhai login

How do you want to login:

> Terminal (login directly inside this terminal)

Browser (open a login page on your browser)

*PAS NÉCESSAIRE GRÂCE À GITPOD !!*

 Utiliser le compte récupéré à l'étape précédente 

# ⚡ Création du Notebook

```
ovhai notebook run conda jupyterlab \  
--name attendee-$STUDENT_ID-yolov8-rock-paper-scissors-notebook \  
--gpu 1 \  
--volume attendee-$STUDENT_ID-data@GRA:/workspace/attendee:RW:cache \  
--volume  
https://github.com/devrel-workshop/101-AI-and-py.git:/workspace/101-ai-lab-sources:RO
```



# Connexion au JupyterLab

```
$ ovhai notebook list | grep attendee- $\$$ STUDENT_ID
```

ID	NAME	STATE	AGE	FRAMEWORK	VERSION
Xxx	<u>attendee-&lt;ID&gt;-yolov8-rock-paper-scissors-notebook</u>	<u>RUNNING</u>	6d	<u>conda</u>	<u>conda-py39-cuda11.8-v22-4</u>

EDITOR	URL
<u>jupyterlab</u>	<u>https://efb3fda1-7b0e-42ed-b6e6-341881071d84.notebook.gra.ai.cloud.ovh.net</u>

**i Utiliser le compte créé précédemment i**





# 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
<a href="#">attendee-backup-ai-lab-notebook</a>	Gravelines	Miniconda- conda-py39-cuda11.8-v22-4	13	1	Private	2mn	user-UtR7ujyZfCgm	Stopped	JupyterLab
<a href="#">attendee-test1-yolov8-rock-paper-scissors-notebook</a>	Gravelines	Miniconda- conda-py39-cuda11.8-v22-4	13	1	Private	2h 30mn	user-jcP2qp858vHD	In service	JupyterLab



# Organisation du Notebook

■ /

Name

■ 101-ai-lab-sources



Répertoire de sources du workshop - RO

■ ai-training-examples



Répertoire de sources d'exemples - RO

■ attendee



Répertoire des données du workshop - RW

- File > New Terminal
- `/workspace/101-ai-lab-sources/  
init_ai_env.sh`

```
Terminal 1 x +
(base) ovh@job-cbe183a9-db28-4239-8b78-e06c8d41cfaf:~$ ls -lart
total 6
-rw-r--r-- 1 ovh ovh 43 Jun 9 12:05 .bash_profile
drwxr-xr-x 2 ovh ovh 1 Jun 9 12:06 .conda
drwxr-xr-x 16 ovh ovh 15 Jun 9 12:06 .miniconda3
-rw-r--r-- 1 ovh ovh 555 Jun 9 12:06 .bashrc
drwx----- 5 ovh ovh 3 Jun 9 12:11 .local
drwxr-xr-x 2 ovh ovh 0 Jun 9 12:11 .ipython
drwxr-xr-x 6 ovh ovh 10 Sep 27 14:47 101-ai-lab-sources
drwxr-xr-x 2 ovh ovh 0 Sep 27 14:47 data
drwxr-xr-x 2 ovh ovh 0 Sep 27 14:47 attendee
drwxr-xr-x 1 root root 4096 Sep 27 14:47 ..
drwxr-xr-x 7 ovh ovh 12 Sep 27 14:48 ai-training-examples
-rw-r--r-- 1 ovh ovh 130 Sep 27 14:48 .workspace.initialized
drwxr-xr-x 11 ovh ovh 12 Sep 27 14:48 .
drwxr-xr-x 3 ovh ovh 2 Sep 27 14:55 .jupyter
(base) ovh@job-cbe183a9-db28-4239-8b78-e06c8d41cfaf:~$
```

# Data

 Source des données : [Roboflow - Rock Paper Scissors SXSU Image Dataset](#)

 Object storage utilisé :

- attendee-\$STUDENT\_ID-data :
  - Zone de travail pour le / la participant·e
  - Dataset
  - Images de tests
  - Répertoires de travail pour l'entraînement
  - Modèle exporté

# YoloV8

 You Only Look Once

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

 Classification / Détection / Segmentation / Détection / Pose

 Entraîné sur le dataset [COCO](#)

 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



# 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

 Bravo vous avez créé votre premier modèle d'intelligence artificielle !! 





# Entraînement avec AI Training



## Instructions

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

<https://ovh.to/z84huKC>



## Rappels

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

# Création de l'image

 Répertoire de travail : **src/training**

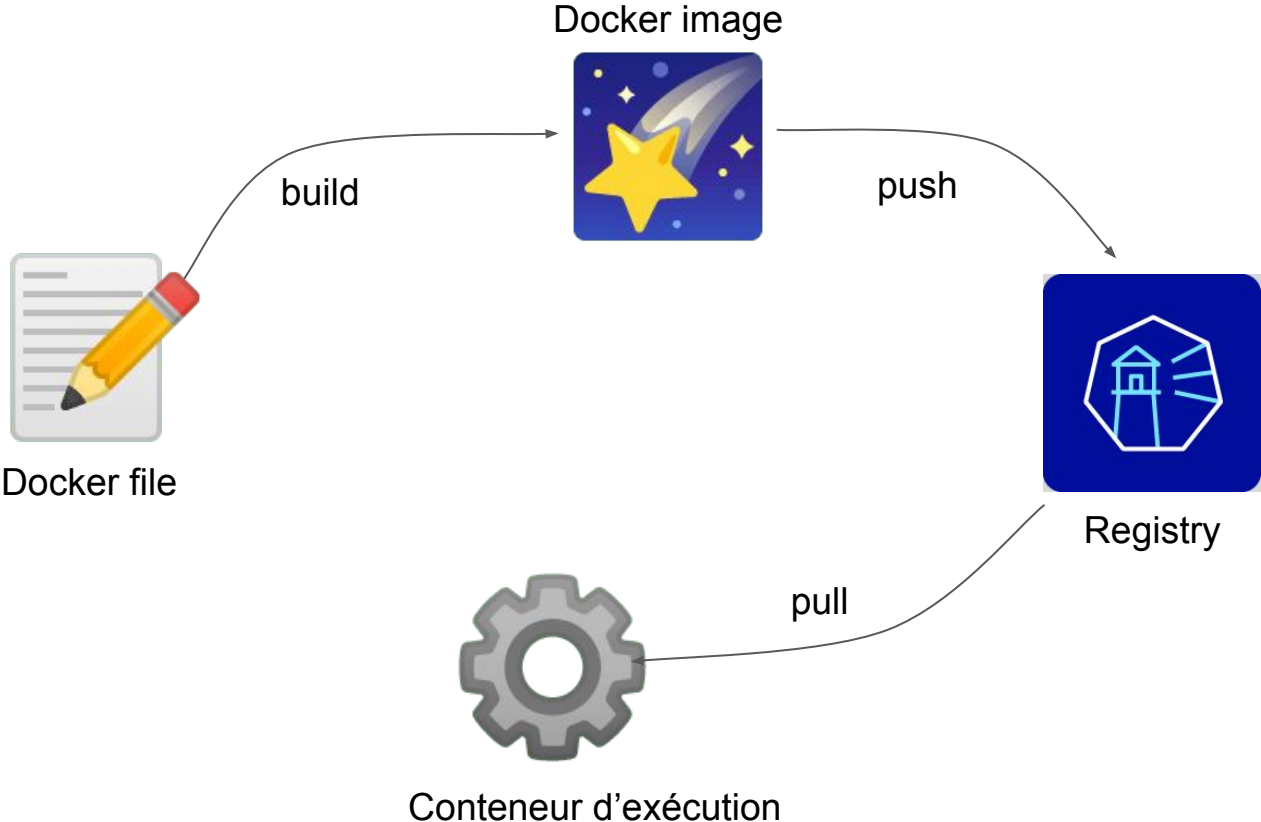
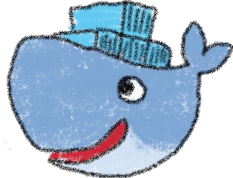
- Dockerfile: le dockerfile pour construire l'image
- Requirements.txt : fichier de gestion des dépendances Python
- Train.py : script Python pour l'entraînement du modèle

 Fabrication de l'image

 (Optionnel) Run / debug localement

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

# Docker



## Création du Job

```
ovhai job run \  
  --name attendee-$STUDENT_ID-yolov8-rock-paper-scissors-training-job \  
  --gpu 1 \  
  --env NB_OF_EPOCHS=50 \  
  --volume attendee-$STUDENT_ID-data@GRA:/workspace/attendee:RW:cache \  
  --unsecure-http \  
  $REGISTRY_NAME/$STUDENT_ID/yolov8-rock-paper-scissors-training-job:1.0.0
```

---

## Plan B

```
ovhai job run \  
  --name attendee-$STUDENT_ID-yolov8-rock-paper-scissors-training-job \  
  --gpu 1 \  
  --env NB_OF_EPOCHS=50 \  
  --volume attendee-$STUDENT_ID-data@GRA:/workspace/attendee:RW:cache \  
  --unsecure-http \  
  $REGISTRY_NAME/backup/yolov8-rock-paper-scissors-training-job:1.0.0
```

## Suivi des logs

```
ovhai job logs -f <job id>
```





# 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 Philippart'. The left sidebar shows a navigation menu with categories: 'Containers & Orchestration' (Managed Kubernetes Service, Kubernetes Load Balancer, Managed Private Registry, Workflow Management), 'AI & Machine Learning' (AI Dashboard, AI Notebooks, AI Training, AI Deploy), and 'Data & Analytics'. The 'AI Training' option is highlighted with a dashed green box. The main content area is titled 'WORKSHOP\_101\_AI / AI Training / Jobs'. It features a 'Launch a new job' button, a search bar, and a 'Filter' button. Below this is a table with the following data:

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



Bravo vous avez entraîné votre premier modèle d'intelligence artificielle !!



Welcome on the 🗑️ 📄 ✂️ game!

Take your picture in real time:



Take Photo

# Application avec AI Deploy



## Instructions

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

<https://ovh.to/4tu9bxc>



# 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 : lab-user / Passw0rd

# ⚡ Création de l'application

```
ovhai app run \  
  --name attendee-$STUDENT_ID-yolov8-rock-paper-scissors-app \  
  --cpu 1 \  
  --default-http-port 8501 \  
  --volume attendee-$STUDENT_ID-data@GRA:/workspace/attendee:RW:cache \  
  --unsecure-http \  
  $REGISTRY_NAME/$STUDENT_ID/yolov8-rock-paper-scissors-app:1.0.0
```

## Plan B

- Copie du modèle entraîné: ovhai bucket object copy attendee-backup-data@GRA  
--container attendee- $\$$ STUDENT\_ID-data best.torchscript best.torchscript
- Exécution de l'image backup:

```
ovhai app run \  
--name attendee- $\$$ STUDENT_ID-yolov8-rock-paper-scissors-app \  
--cpu 1 \  
--default-http-port 8501 \  
--volume attendee- $\$$ STUDENT_ID-data@GRA:/workspace/attendee:RW:cache \  
--unsecure-http \  
 $\$$ REGISTRY_NAME/backup/yolov8-rock-paper-scissors-app:1.0.0
```

# Accéder à l'application

```
$ ovhai app list | grep lab/$STUDENT_ID
```

ID	STATE	IMAGE	REPLICAS	SCALING_STRATEGY
<u>&lt;app id&gt;</u>	<u>RUNNING</u>	<u>xxxxxx/lab/&lt;ID&gt;/yolov8-rock-paper-scissors-app:1.0.0</u>	<u>1</u>	<u>fixed (1)</u>

```
$ ovhai app <app id>
```

Status:

State: RUNNING

Available Replicas: 0

Url: <https://<app id>.app.gra.ai.cloud.ovh.net>

Info Url: <https://ui.gra.ai.cloud.ovh.net/app/<app id>>

Monitoring Url: <https://monitoring.gra.ai.cloud.ovh.net/d/app?var-app=<app id>&from=1695898895255>



## Suivi des logs

```
ovhai app logs -f <app id>
```



# 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 **1**

 Bravo vous avez créé votre première application basée sur l'intelligence artificielle

!! 



Take away / Next ?



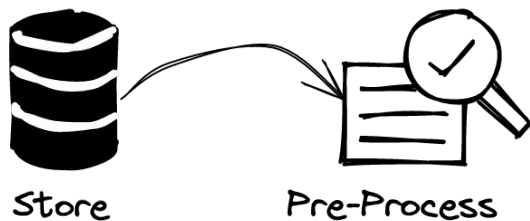
# Un workflow typique dans l'IA



Store

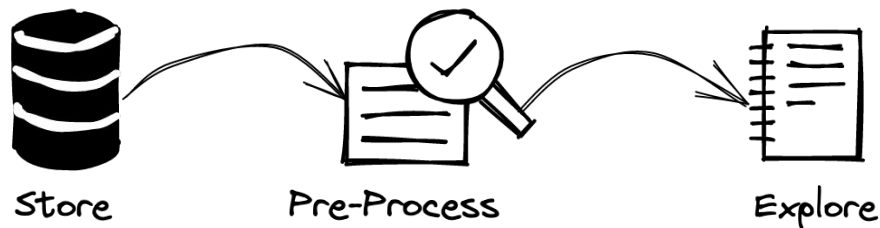
- 🕒 Il faut une très grande quantité de données
- 🕒 La plupart du temps la donnée est brute
- ⚠️ Attention aux coûts (stockage, lecture / écriture)

# 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

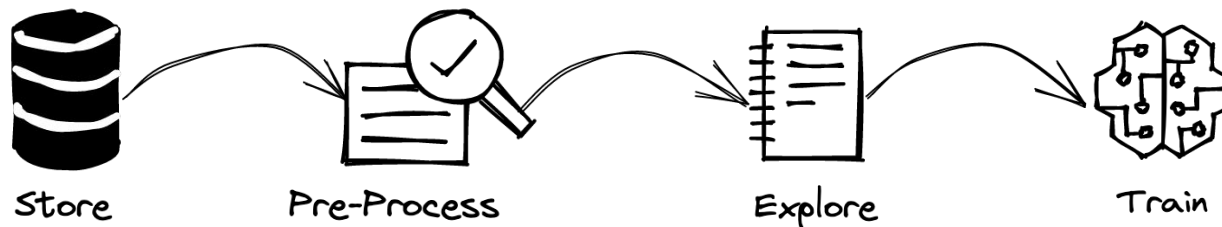
 Basé sur des opérations mathématiques complexes

 Réseaux de neurones artificiels

 Fait par les Data Scientist

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

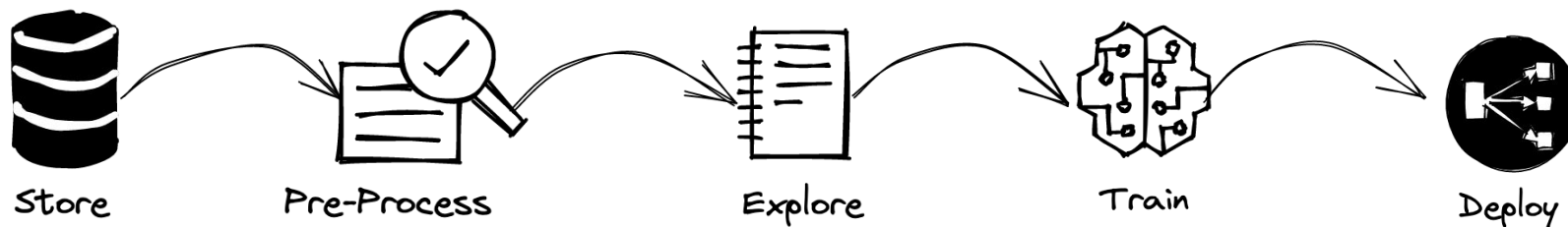
# Un workflow typique dans l'IA



- ✨ 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)

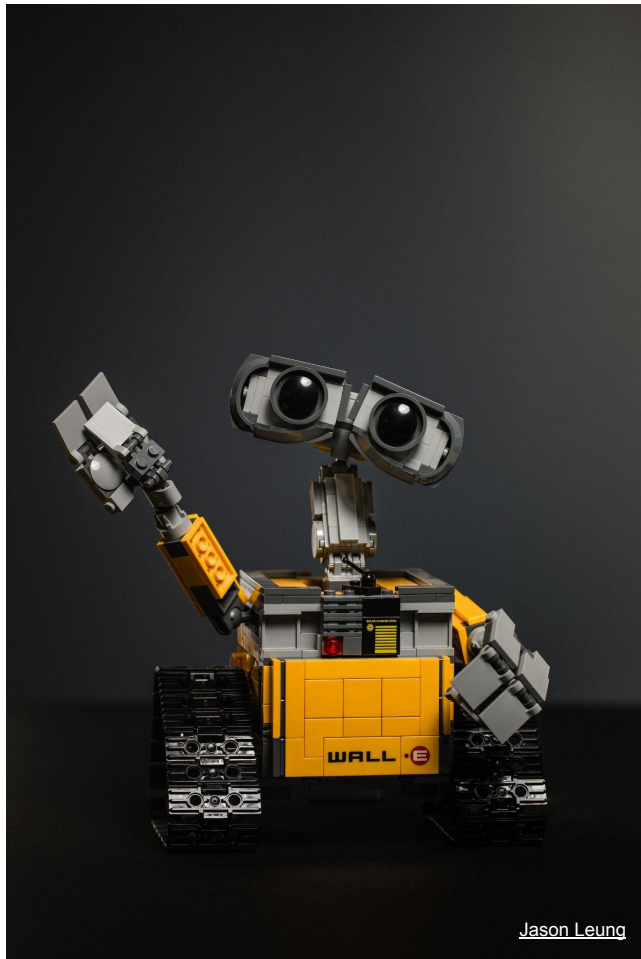


# Un workflow typique dans l'IA



 Utilise le modèle pour faire l'inférence

 Peut exposer une API ou une interface utilisateur



Jason Leung

# 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 statistiques

🤔 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"

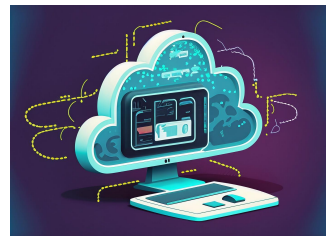
 Avoir de la performance avec moins de consommation

# La diversité des CDE

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



Gitpod



GitHub Codespaces

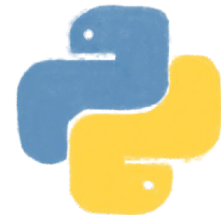


Jetbrains Space



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



Golearn



Langchain pour Java



Tensorflow js



Slides

<https://ovh.to/ofPmXWY>

# Merci !!!!



Feedbacks

<https://ovh.to/FEM3HSz>



Code source

<https://ovh.to/ofPmni6>



# OVHcloud

OVHcloud recrute :

<https://ovh.to/4tu91L6>





23 novembre 2023  
Maison de la Mutualité

**INSCRIVEZ-VOUS  
DÈS MAINTENANT**



Plus d'informations sur  
[summit.ovhcloud.com](https://summit.ovhcloud.com)

# Ressources (liens, ...)

# Liens

 [Documentations](#) OVHcloud univers IA

 [Repository GitHub](#) du workshop

 <https://unsplash.com/>

 <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://www.youtube.com/@MachineLearnia>

 <https://gitpod.io>

