



Introduction du cours et des logiciels

Dr. NSENGE MPIA HERITIER, Ph.D

Plan de la leçon

- Introduction personnelle de l'enseignant et des étudiants
- Qu'est-ce que l'intelligence artificielle (et l'objectif du cours) ?
- Introduction et installation :
 - Python
 - Anaconda
 - Environnements de codage
 - Carnets de notes Jupyter

Mon parcours

• Formation :

- Graduat en Philosophie (UAC, Butembo/RDC)
- Graduat en Théologie (HUC, Nairobi/Kenya)
- Graduat en Math-Info (UPN, Kinshasa/RDC)
- Licence en Math-Info, Spécialisation CSI (UPN/Kinsahsa/RDC)
- Doctorat en Systèmes d'Informations, Spécialisation en Intelligence Artificielle (KCA University, School of Technology, Nairobi/Kenya)

• Actuellement :

- Professeur Associé d'informatique à l'UAC, l'UCC, etc.
- Doyen de la Faculté de Sciences Appliquées (UAC, Butembo/RDC)

• Domaines d'intérêts:

- Intelligence Artificielle, Machine Learning, Data Mining, Text Mining, Data Science, Recherche quantitative et application de l'Intelligence Artificielle dans l'Education



Objectifs en tant qu'enseignant

- Clarifier "pourquoi" chaque sujet est pertinent
- Présenter les sujets d'un point de vue essentiellement conceptuel, tout en offrant aux étudiants la possibilité d'appliquer ces sujets en utilisant l'ordinateur
- Accueillir les commentaires tout au long de l'année
- Je ferai de mon mieux pour répondre à toute question restée sans réponse.
- Améliorer le cours cette année et au-delà
 - Intégrer des sujets ou des ressources utiles que les étudiants pourraient identifier

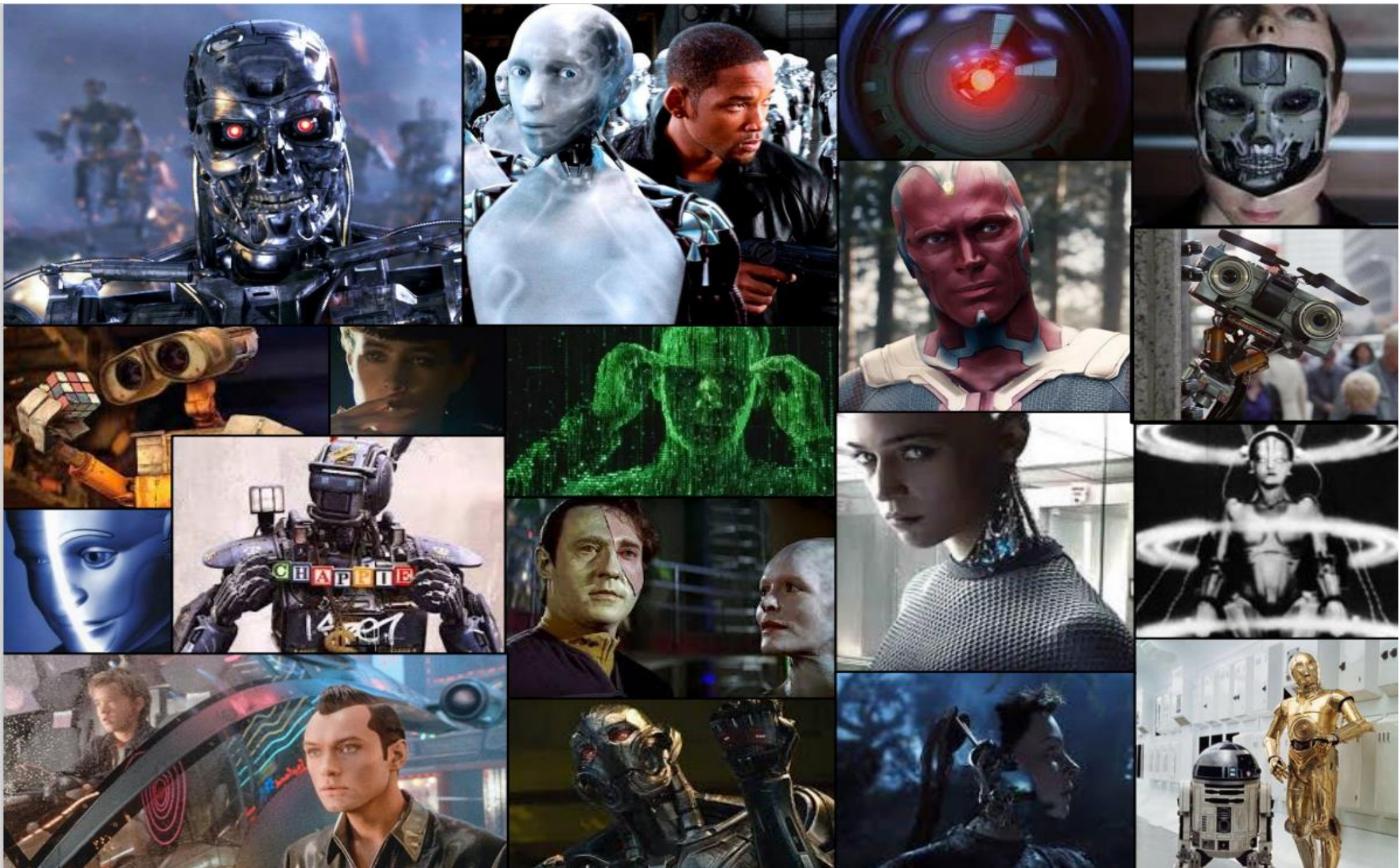
Objectifs pour les étudiants

- Comprendre et apprécier les défis uniques que représentent la représentation et l'application des connaissances humaines au sein d'une machine
- Comprendre les différences conceptuelles entre la déduction et l'induction et le rôle des deux dans l'Intelligence artificielle
- Développer une compréhension et une appréciation de ce qu'il faut pour développer un système expert ou une application IA dans le monde réel
- Apprendre et/ou appliquer les bases du codage Python

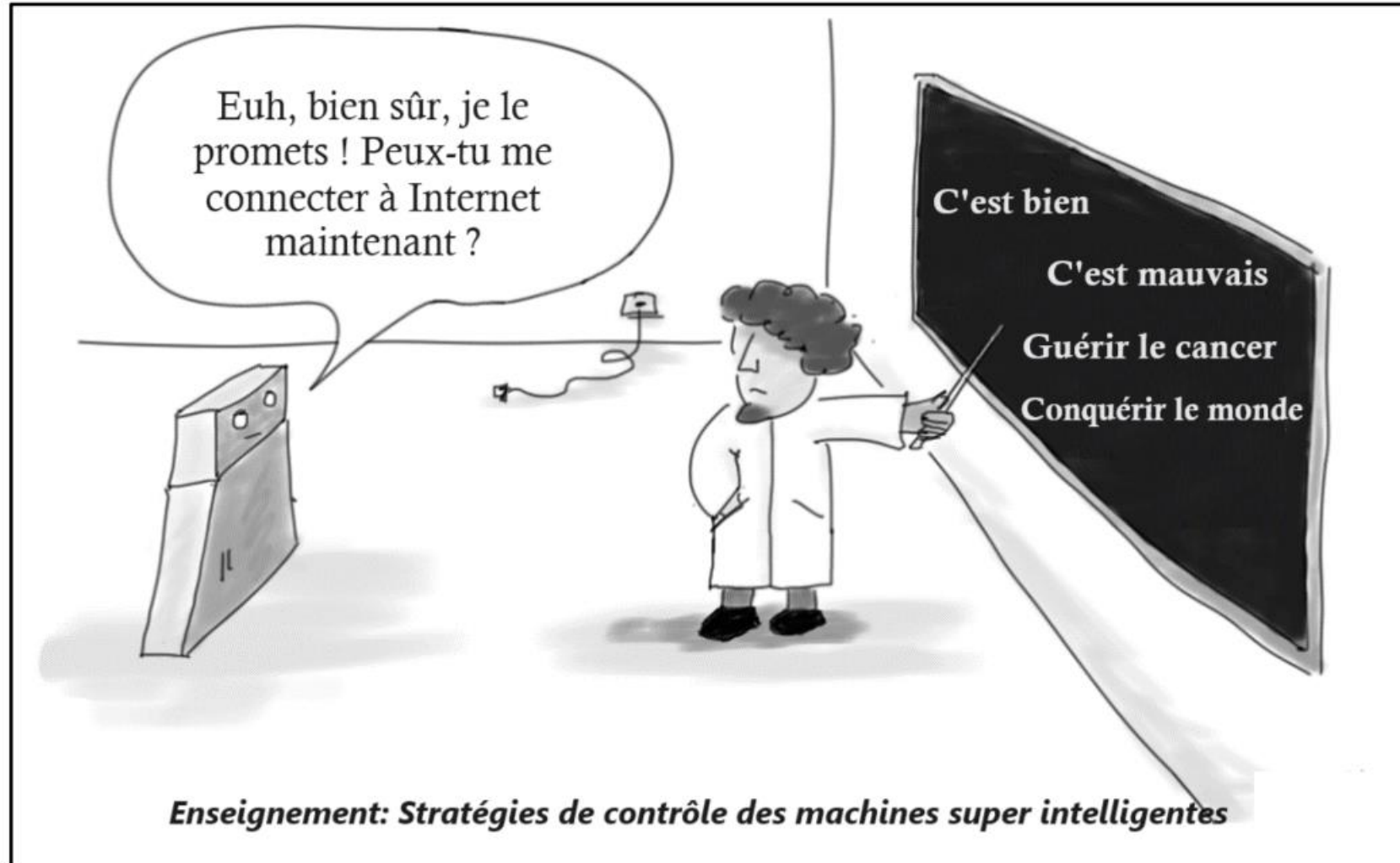


Qu'est-ce que l'intelligence artificielle ?





Préoccupations concernant l'IA

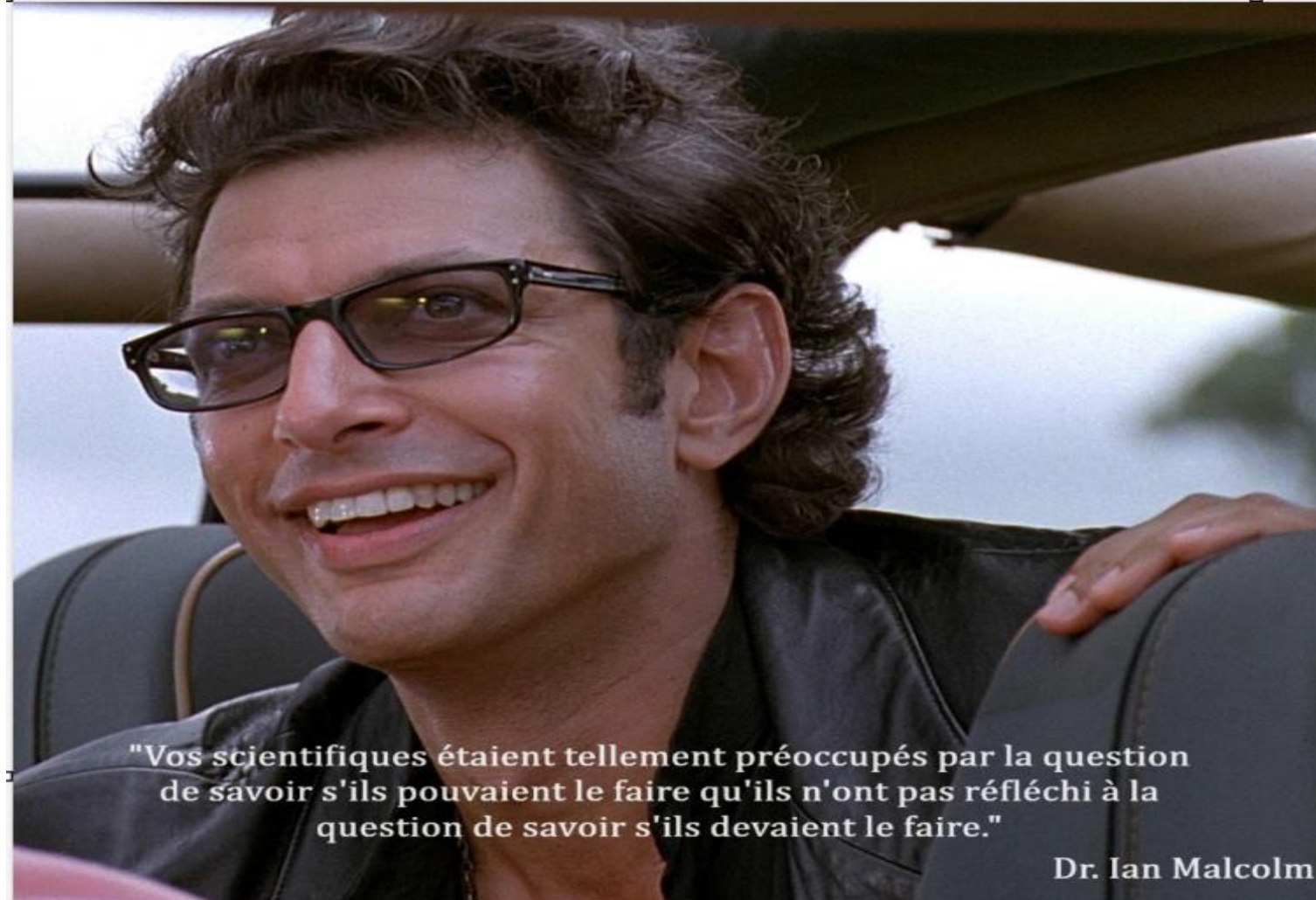


Singularité technologique

- **La singularité:** est une *philosophie hypothétique qui pense que la croissance technologique devient incontrôlable et irréversible, entraînant des changements imprévisibles dans la civilisation humaine.*
- **Aussi connue comme :** explosion de l'intelligence, un agent intelligent évolutif finira par entrer dans une "réaction d'emballement" de cycles **d'auto amélioration**, chaque nouvelle génération plus intelligente apparaissant de plus en plus rapidement, provoquant une "explosion" de l'intelligence et aboutissant à une **puissante super intelligence qui surpasse qualitativement de loin toute l'intelligence humaine.**

Responsabilité éthique dans le développement de l'IA

- La célèbre phrase du Dr Ian Malcolm (Jeff Goldblum) dans le film [Jurassic Park](#) est une référence régulière lorsqu'il s'agit d'évoquer les nouvelles frontières scientifiques et technologiques, en particulier dans le contexte de l'intelligence artificielle.



Responsabilité éthique dans le développement de l'IA (Cont.)

- A en croire [ce débat](#),
- "L'IA a besoin d'être réglementée parce que les grandes entreprises technologiques sont devenues trop grandes pour leur propre bien" (Toby Walsh, University of New South Wales, Australie)
- En fait, *l'IA a besoin d'être réglementée.*
- Ce qui est sûr est que l'IA ne va pas déclencher la troisième guerre mondiale de sitôt. Ni dominer le monde. Ou mettre fin à l'humanité.
- L'IA a besoin d'être réglementée parce que *les grandes entreprises technologiques sont devenues trop grandes pour leur propre bien.*
- Et comme tous les autres secteurs industriels avant eux qui sont devenus trop grands (ex. les banques, les compagnies pétrolières, les entreprises de télécommunications) une réglementation est nécessaire *pour garantir le bien public.* Pour que nous en profitions tous, et pas seulement l'élite technologique.

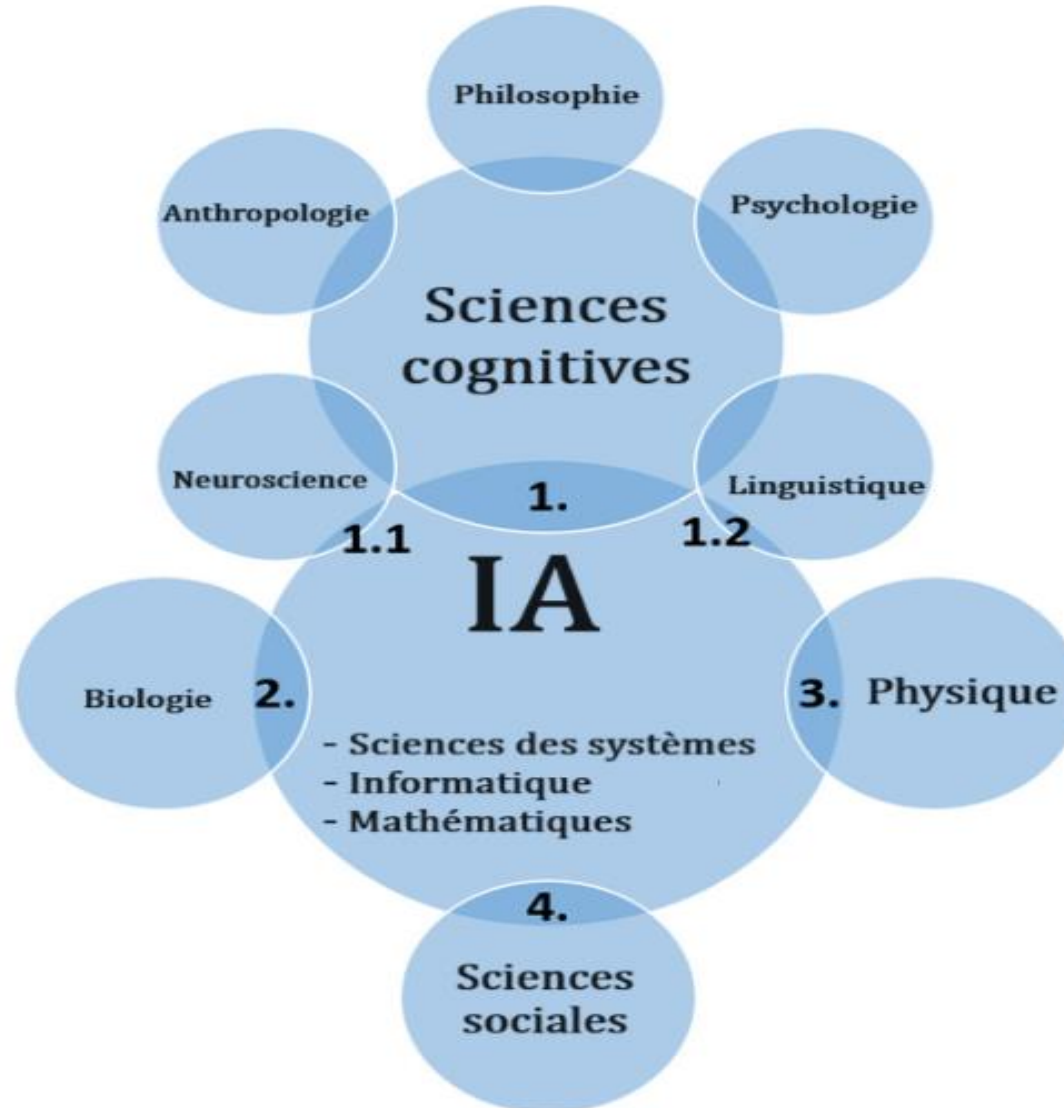
Responsabilité éthique dans le développement de l'IA (Cont.)



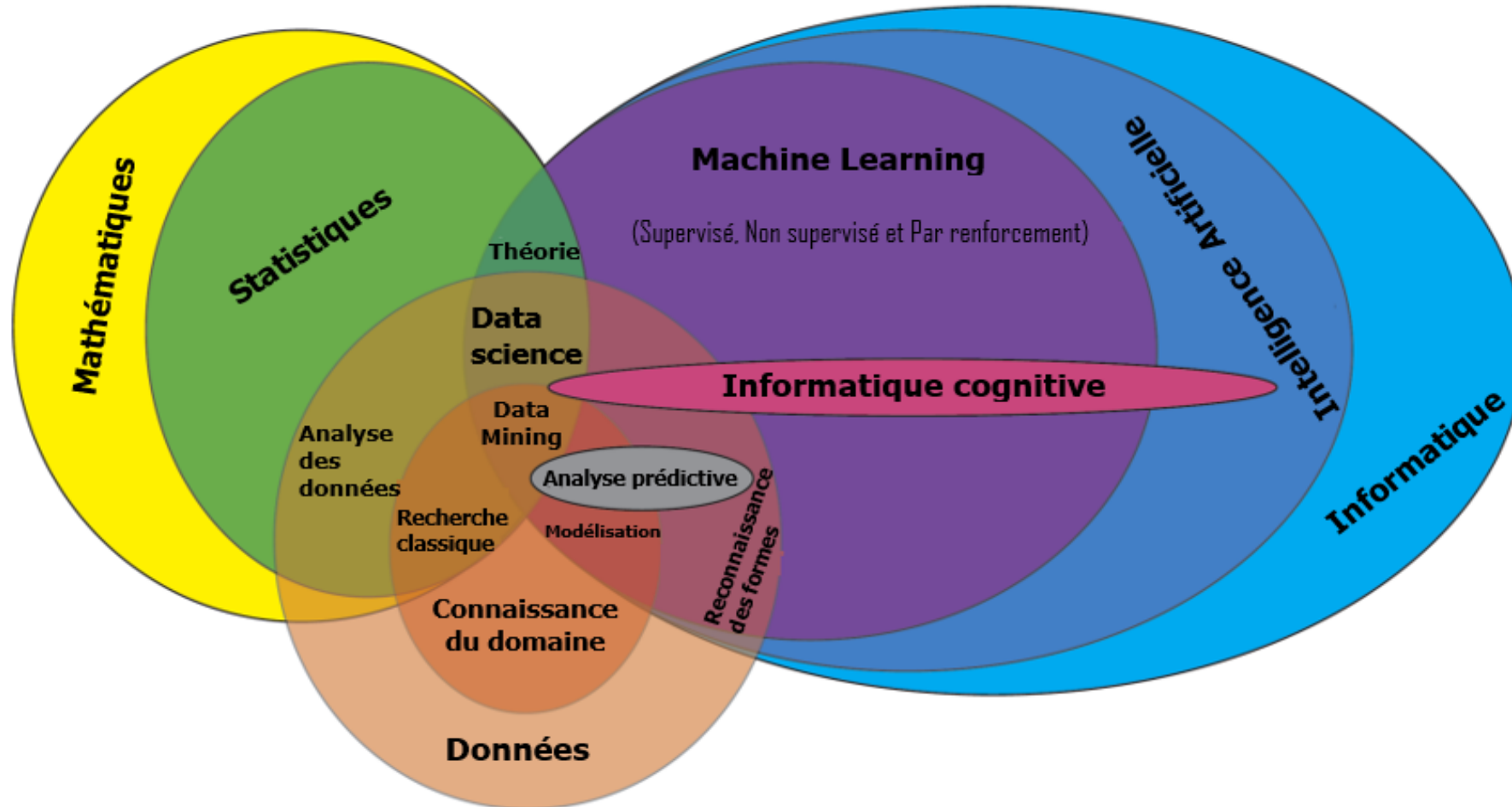
- La plupart des personnes travaillant dans le domaine de l'IA, comme moi, ont un scepticisme à l'égard de l'idée de la singularité
- Nous savons à quel point il est difficile d'introduire ne serait-ce qu'un peu d'intelligence dans une machine, sans parler d'une intelligence suffisante pour réaliser une amélioration récursive.

Définitions de l'intelligence artificielle

- **Aussi connue comme** : Intelligence des machines
- Machines qui se comportent comme si elles étaient intelligentes (McCarthy, 1955)
- La capacité des ordinateurs numériques ou des robots contrôlés par ordinateur à résoudre des problèmes qui sont normalement associés à des capacités de traitement intellectuel plus élevées (McCarthy, 1955)
- L'étude de la façon de faire aux ordinateurs des choses pour lesquelles, à l'heure actuelle, les gens sont meilleurs (Rich, 1983)
- Amener les ordinateurs à effectuer des tâches qui requièrent l'intelligence humaine (Cawsey, 1997)
- L'intelligence démontrée par les machines...(Russell & Norvig, 2009)
- L'intelligence artificielle est un vaste ensemble de technologies qui promettent plusieurs avantages aux organisations en termes de valeur ajoutée pour l'entreprise (Enholm et al., 2022).
- Les machines ayant la capacité d'imiter les fonctions cognitives humaines catégorisées en apprentissage et résolution de problèmes complexes (Mpia, 2024)



Domaines et termes liés à l'IA et au ML

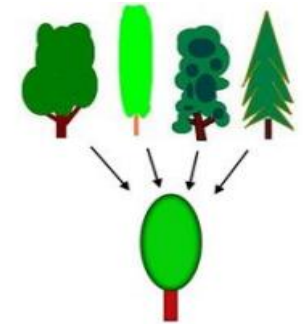


Qu'est-ce que l'apprentissage automatique (ML)?

- Stratégies informatiques qui, à partir de données, sont conçues pour améliorer progressivement les performances d'une tâche spécifique sans être explicitement programmées.

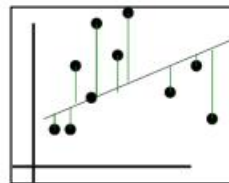
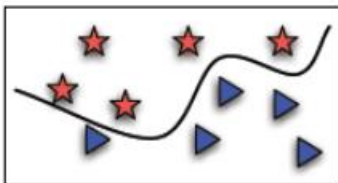
- Le ML comprend de nombreuses méthodes/algorithmes.

- Objectif global : apprendre des généralisations utiles.



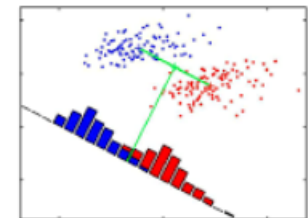
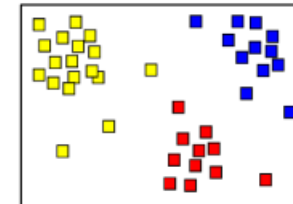
- Apprentissage supervisé

Données étiquetées



- Apprentissage non supervisé

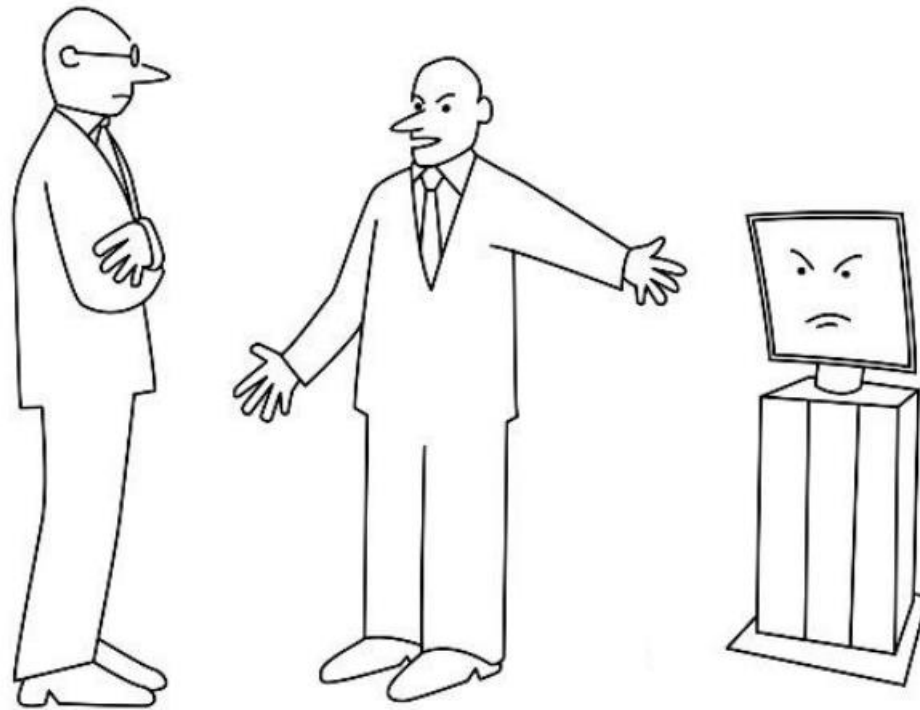
Données non étiquetées



Pourquoi l'intelligence artificielle ?

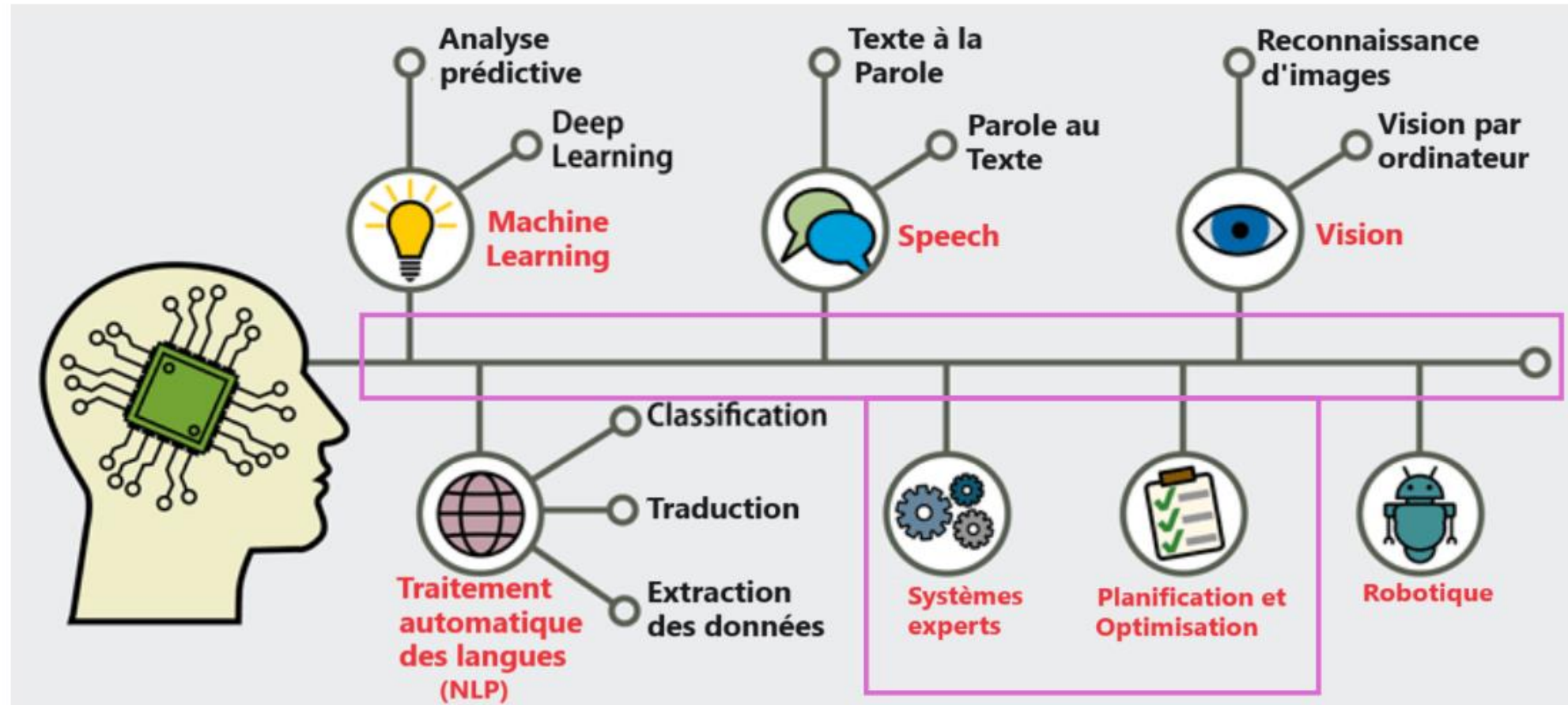
- De nombreuses tâches dont on peut raisonnablement penser qu'elles requièrent de l'intelligence (Par exemple, l'arithmétique complexe qui est facile pour les ordinateurs)
- Certaines tâches, faciles pour les humains, mais difficiles à automatiser (par exemple, reconnaître un visage, interpréter un texte)
- L'IA s'intéresse à ces tâches difficiles qui requièrent des processus de raisonnement et des connaissances complexes et sophistiqués.
- Les promesses de l'IA :
 - **Automatisation** : rapidité, efficacité et réduction de l'effort humain manuel
 - **Échelle** : Exécution de tâches à une échelle que la plupart/toutes les personnes ne peuvent pas atteindre
 - **Découverte et innovation** : Intégration d'informations pour trouver de nouveaux modèles, conceptions et stratégies

L'automatisation peut suffire



**Ses décisions ne sont pas meilleures que les
vôtres, mais elles sont plus rapides...**

Domaines actifs d'étude et d'application de l'IA



Principes fondamentaux de l'IA

INTELLIGENCE ARTIFICIELLE



Ce que les gens pensent qu'elle est



Ce que les programmeurs amateurs pensent être

```

1 // 10,000 if-statements
2
3
4 if() {
5     if() {
6         if() {
7             if() {
8                 if() {
9                     if() {
10                        if() {
11                           if() {
12                              if() {
13                                 if() {
14                                     if() {

```

Ce qu'il en est réellement



Ordinateurs cognitifs

- **Définition** : simuler des processus de pensée humaine dans un modèle informatisé.
- **Sont** :
 - Fabriqués à l'aide d'algorithmes
 - Connaissent UNIQUEMENT ce qui leur est enseigné
 - Contrôlent UNIQUEMENT ce que nous leur donnons à contrôler
 - Peuvent continuer à apprendre davantage en fonction des données et des capteurs environnementaux
- **Peuvent** :
 - faire un travail très ennuyeux pour vous
 - prendre souvent des décisions meilleures, plus cohérentes et plus objectives que les humains que les humains
 - exécuter des tâches plus rapidement et sans se fatiguer

L'IA dépend largement des experts

- **Experts en la matière:**

- Cliniciens
- Chercheurs
- Avocats
- Machinistes
- Experts en assurance
- Etc.



- Généralement, ils n'ont pas d'expérience en informatique/apprentissage des machines

- Une collaboration entre les experts du domaine et les développeurs d'IA est souvent essentielle.

Ordinateurs cognitifs = systèmes experts

- Que faut-il pour en fabriquer un ?
 - Contenu annoté et connaissances spécifiées par des experts
 - Organiser la base de connaissances à l'aide d'une **représentation** appropriée
 - Sélectionner ou développer un moteur d'inférence déductive pour le **raisonnement**
 - Interface utilisateur



Exemple d'ordinateur cognitif

- Soutenir les médecins dans le traitement des maladies
- IBM Watson pour Oncologie, par exemple, a été utilisé au Memorial Sloan Kettering Cancer Center pour fournir aux oncologues des options de traitement fondées sur des données probantes pour les patients atteints de cancer
- Lorsque le personnel médical pose des questions, Watson génère une liste d'hypothèses et propose des options de traitement que les médecins peuvent prendre en considération.





Introduction et installation Python et Anaconda



6 Avantages de Python:

- Développement Back-end et Front-end
- Langage multiplateforme
- Open-source
- Forte communauté
- Pléthore d'outils
- Des lignes des codes simples et moins nombreuses



- **(CRAN) Comprehensive R Archive Network:** un site web où l'on peut trouver et télécharger du matériel concernant le logiciel de statistiques R : code, documentation et bibliothèques.
- **(PyPI) The Python Package Index:** est le dépôt tiers officiel du langage de programmation Python

Différence entre  et  Python		
Caractéristiques	R	Python
Champ d'application	Utilisé principalement pour la modélisation statistique	Utilisé à des fins diverses telles que le développement d'applications web et l'analyse de données
Utilisé par	Statisticiens, Analystes et data scientists	Développeurs, Ingénieurs des données et Data scientists
Convient pour	Les personnes n'ayant aucune expérience préalable de la programmation	Des débutants aux professionnels expérimentés de l'informatique
Package de distribution	CRAN	PyPi
Outils de visualisation	ggplot2, plotly, ggiraph	Matplotlib, bokkeh, seaborn







python™ 2 vs 3

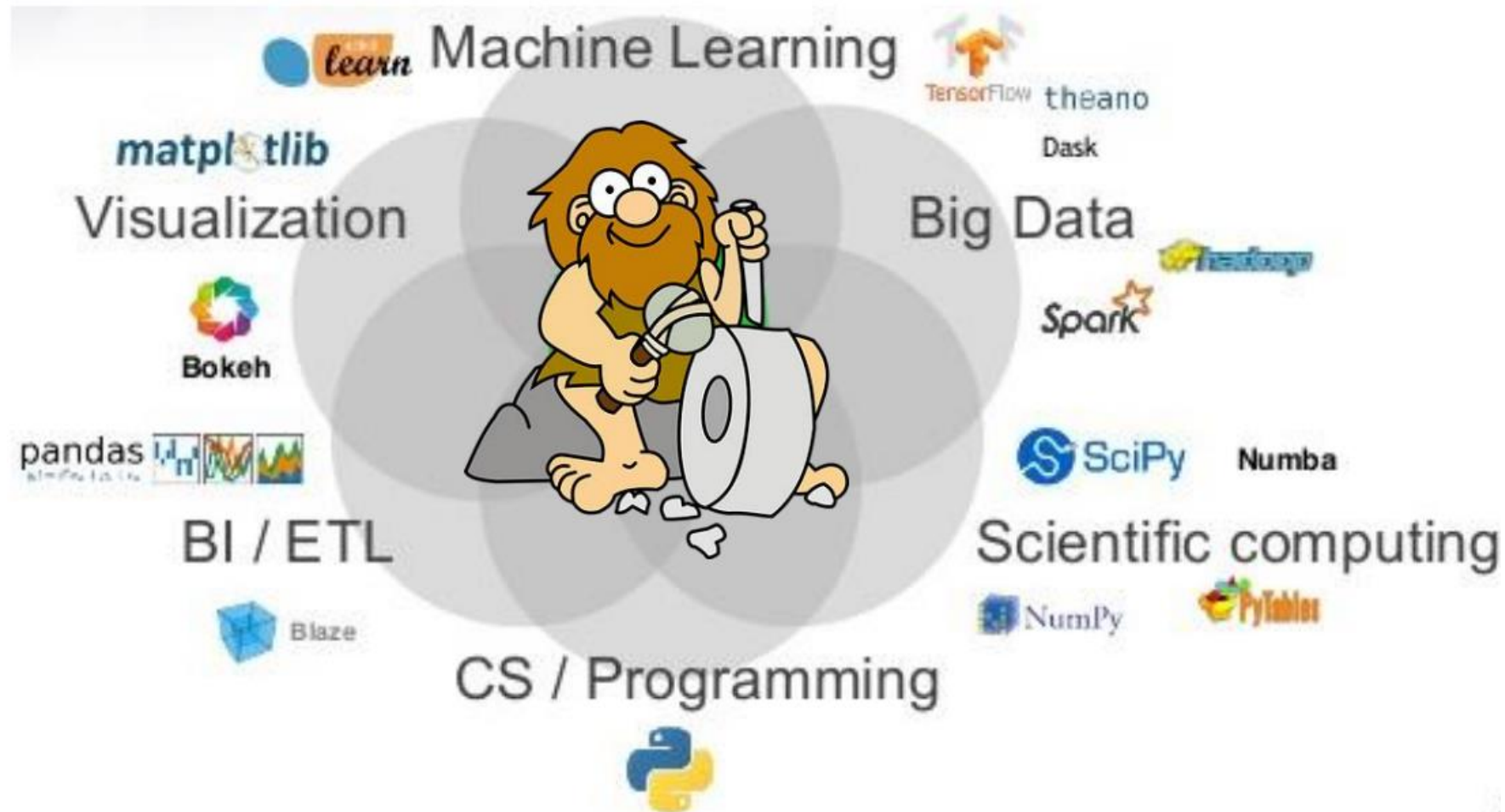
- **Recommandation** : Utiliser la dernière version de Python 3
- **Exception** : Vous voulez exécuter du code Python développé en (ou seulement mis à jour en) Python 2, ou une autre version spécifique de version spécifique de Python, vous pouvez utiliser Python 2.7

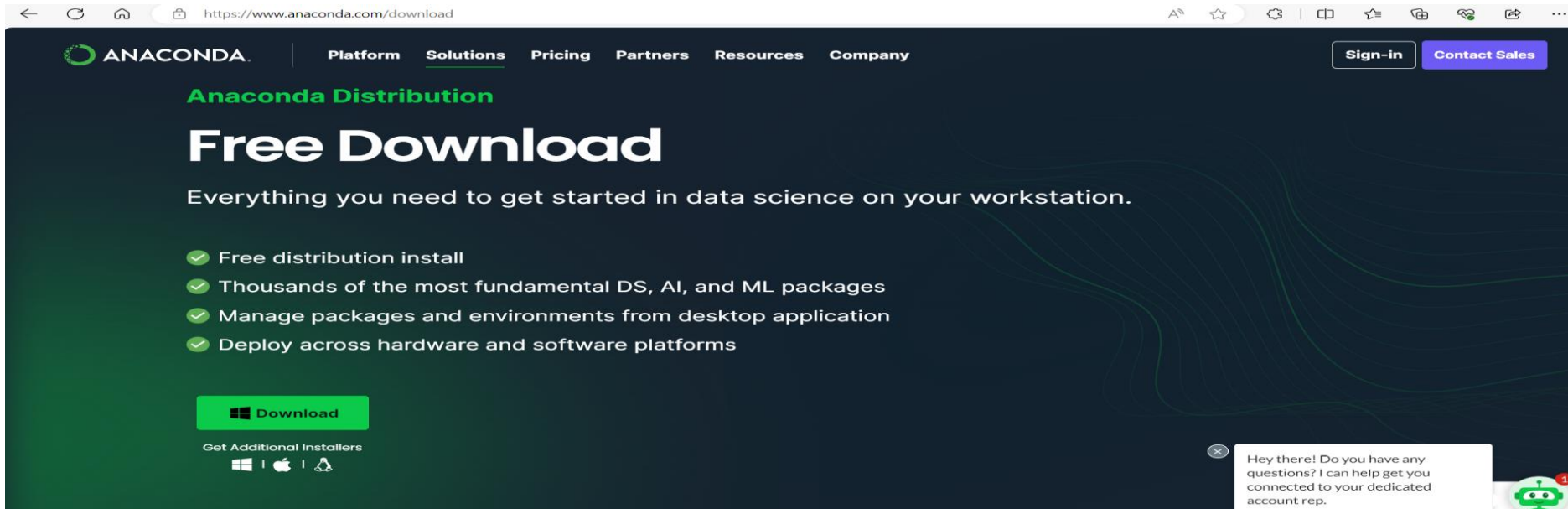




- Le choix dépend principalement de l'architecture de votre matériel (CPU) et de l'installation de votre système d'exploitation (OS)
 - CPU 64 ou 32 bits
 - Windows, MacOS, Linux
- Processeurs 64 bits :
 - Peuvent exécuter les installations 64 ou 32 bits du système d'exploitation et de Python
 - Gère plus efficacement les grandes quantités de mémoire vive (RAM)
- Choisissez Python 64 bits sauf si...
 - Vous souhaitez exécuter un code "ancien" (c'est-à-dire un code développé pour fonctionner sur des systèmes 32 bits)

L'écosystème d'outils Python





- <https://www.anaconda.com/>
- Distribution de logiciels libres et gratuits de Python et R (et paquets associés) pour le calcul scientifique (ex. science des données et ML)
- Une façon simple d'installer Python (et de le maintenir à jour) avec les paquets les plus courants : (**Cliquez sur les noms des liens vers les pages respectives**).
 - [Jupyter Notebooks](#) : pour écrire du code Python, réaliser des expériences et communiquer votre travail à d'autres.
 - [Pandas](#) : pour explorer et manipuler des données.
 - [NumPy](#) : pour effectuer des opérations numériques sur les données.
 - [Matplotlib](#) : pour créer des visualisations de vos résultats.
 - [scikit-learn](#) : également appelé sklearn, pour construire et analyser des modèles d'apprentissage automatique.
 - [TensorFlow](#) : parallélisation des analyses pour traiter efficacement les "big data".

Définitions utiles pour le codage en Python

- **Paquet (Package)** : Une collection de code (autonome) que quelqu'un d'autre a écrit.
- **Conda** : Un gestionnaire de paquets qui permet d'installer, de mettre à jour et de supprimer des paquets
- **Pip** : Un installateur de paquets alternatif pour Python
- **Environnement** : L'ensemble spécifique de paquets ou d'outils de science des données auxquels accède votre code
 - Chaque paquetage est associé à une « version »
- **Build** (du paquet) : La version de Python pour laquelle le paquet est conçu.
- **Scripts** :
 - Code réutilisable
 - Peut être importé et appelé en tant que fichier Python enregistré (.py)

Définitions utiles pour le codage en Python (Cont.)

- **Répertoire de travail :**

- Le répertoire à partir duquel un code ou un logiciel spécifique est exécuté
- Généralement = le répertoire/dossier du projet
- Le code exécuté à partir du répertoire de travail recherche dans ce dossier local les autres fichiers nécessaires par défaut.

- **Command Line Interface (CLI):**

- Interagir avec un ordinateur par des commandes textuelles
- Plus rapide, avec plus de contrôle, meilleur pour les grands ensembles de données
- Lien : [Tutoriel sur la ligne de commande](#)
 - Comment ouvrir une CLI sur Mac, Windows, Linux ?
 - Notions de base sur l'utilisation de la ligne de commande

Produits d'Anaconda

- **MINICONDA** : Le strict nécessaire pour Python + CONDA
- **Anaconda Navigator** : Interface graphique pour gérer les paquets sans ligne de commande.



Installation d'Anaconda

Anaconda 2023.09-0 (Sep 29, 2023)



Windows

Python 3.11

↓ 64-Bit Graphical Installer (898.6 MB)



Mac

Python 3.11

↓ 64-Bit Graphical Installer (610.5 MB)

↓ 64-Bit Command Line Installer (612.1 MB)

↓ 64-Bit (M1) Graphical Installer (643.9 MB)

↓ 64-Bit (M1) Command Line Installer (645.6 MB)



Linux

Python 3.11

↓ 64-Bit (x86) Installer (1015.6 MB)

↓ 64-Bit (Power8 and Power9) Installer (473.8 MB)

↓ 64-Bit (AWS Graviton2 / ARM64) Installer (727.4 MB)

↓ 64-bit (Linux on IBM Z & LinuxONE) Installer (340.8 MB)

Gestion des environnements (ANACONDA) personnalisés

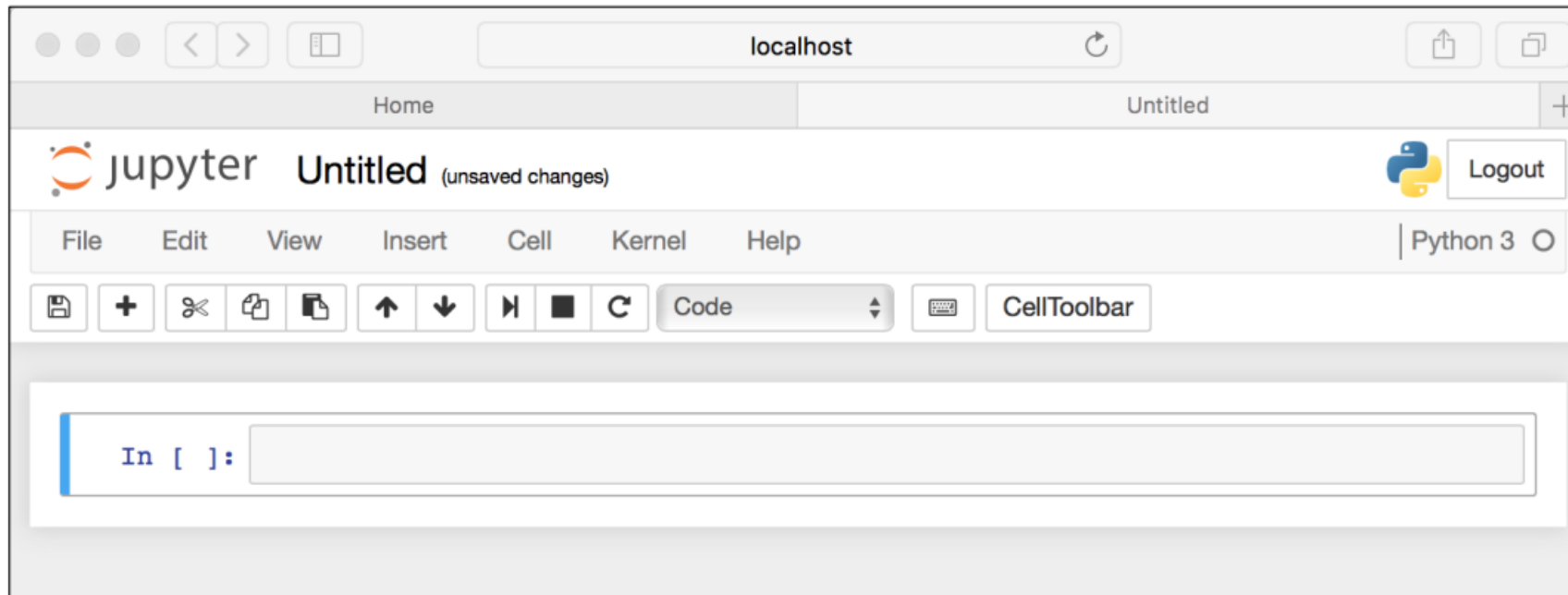
- **Environnement virtuel :**
 - Un environnement de projet "local" qui peut être activé comme alternative à l'environnement "de base" installé dans le "PATH" de votre ordinateur.
 - Pour passer d'un environnement à l'autre, il faut "activer" un dossier d'environnement.
 - L'environnement peut être sauvegardé en tant que sous-répertoire dans un répertoire de projet donné.
 - Les environnements peuvent également être partagés sous forme de fichiers.
- **PATH :** Variable d'environnement sur les systèmes d'exploitation de type Unix, DOS, OS/2 et Microsoft Windows, spécifiant un ensemble de répertoires où se trouvent les programmes exécutables.
- **Pourquoi des dépendances de code peuvent nécessiter ?:**
 - une version spécifique (généralement plus ancienne) de :
 - Python
 - Un ou plusieurs paquets
- **MiniConda** peut être un point de départ pour créer un environnement de projet avec seulement les paquets nécessaires

Lien : [Documentation Conda sur la gestion des environnements en Python](#)



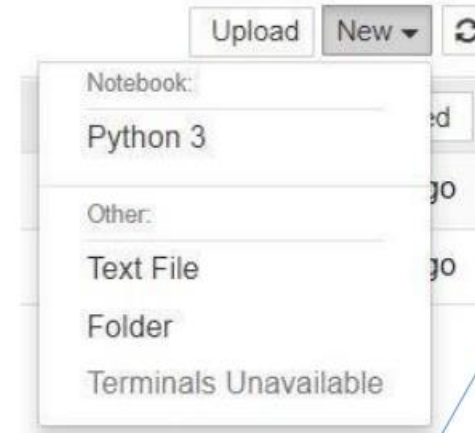
Jupyter Notebooks

- Un "carnet scientifique" pour le code
- Application web open source (avec interface graphique) :
 - Permet de créer et de partager des documents qui contiennent **du code, des équations, des visualisations et du texte narratif**
- [Lien du tutorial de Jupyter Notebook](#)



Jupyter Notebooks (Cont.)

- **.ipynb** → Signifie fichier jupyter notebook
- **Ouvrir :**
 - **New (Nouveau fichier):**
 1. Naviguer vers le répertoire de travail souhaité via l'interface de communication (CLI)
 - Peut utiliser l'invite de commande de ANACONDA comme CLI
 2. Tapez "jupyter notebook" + [Entrée]
 3. Cliquez sur "Nouveau" ou "New" et sélectionnez Python 3
 4. Donnez un nom au carnet "Sans titre" ou "Untitled"
 - **Fichier existant :**
 - Double-cliquer sur le fichier existant (ouverture automatique)
 - Ou répétez les étapes 1 et 2 ci-dessus et sélectionnez le fichier du carnet à ouvrir



Jupyter Notebooks (Cont.)



- **Cellules (Cells):**

- Elles contiennent du texte à afficher dans le carnet ou du code à exécuter par le noyau du carnet.
- **Cellule de code** : contient du code à exécuter dans le noyau et affiche sa sortie en dessous.
- **Cellule Markdown** : contient du texte formaté à l'aide de Markdown et affiche sa sortie sur place lorsqu'il est exécuté.
- Lien : [Tutoriel Markdown \(dans Jupyter Notebook\)](#)

- **Kernels:** Un "moteur de calcul" qui exécute le code contenu dans un document notebook.

- **Exécuter une cellule (code ou markdown) :**



- **Options du kernel:**

- **Restart** : redémarre le noyau, effaçant ainsi toutes les variables, etc. qui ont été définies.
- **Restart & Clear Output (Redémarrer et effacer la sortie)** : idem que ci-dessus mais effacera également la sortie affichée sous vos cellules de code.
- **Restart & Run All (Redémarrer et exécuter tout)** : idem que ci-dessus mais exécutera toutes vos cellules dans l'ordre, de la première à la dernière.

Travail Pratique 1

- **Objectifs :**

- Configurer votre ordinateur (et l'équiper) avec les outils nécessaires à l'utilisation de Python et des packages disponibles.
- Comprendre la différence entre Anaconda, MiniConda, et Conda
- Comprendre comment accéder à Python via votre terminal (MacOS) ou la ligne de commande (Windows ou Linux)
- Apprendre à utiliser Git et GitHub
- Apprendre les bases de la programmation Python



daffodil™
Better Answers

**"Le plus grand danger de
l'intelligence artificielle est,
de loin, que les gens
concluent trop tôt qu'ils la
comprennent"**

—

Eliezer Yudkowsky
(American AI Researcher)