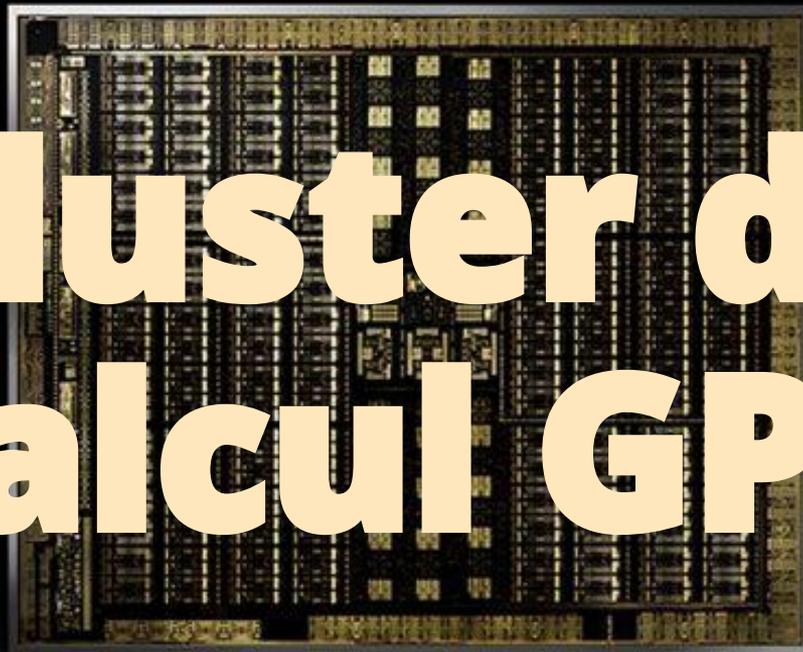




Cluster de Calcul GPU



© NVIDIA



Plan

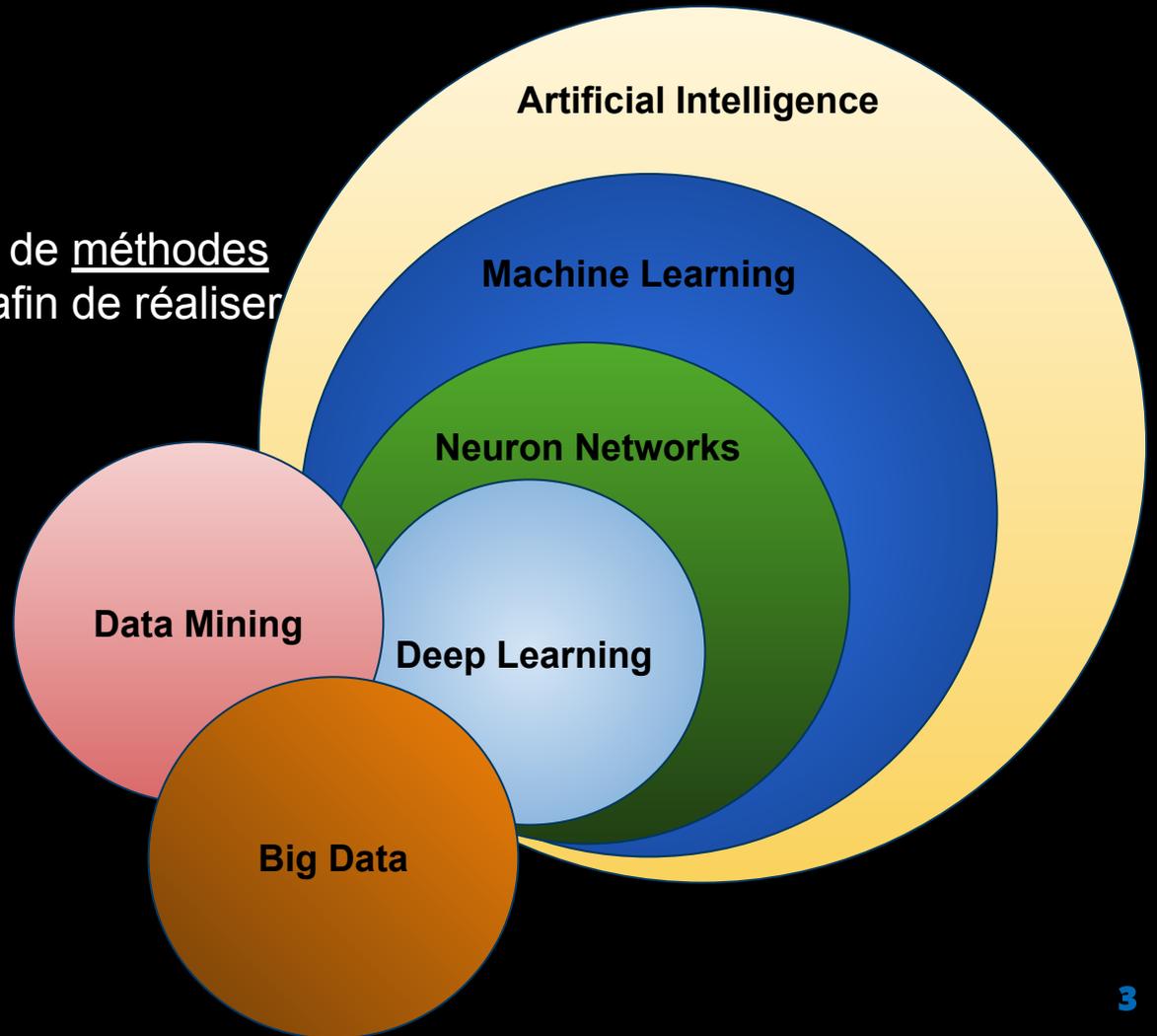
1. Courte introduction au Machine Learning et Deep Learning (slides 3 à 11)
2. Exemples de projets scientifiques basés sur le Deep Learning (slides 12 à 19)
3. Les GPU et différences avec les CPU (slides 20 à 27)
4. Le cluster GPU OVSQ/LATMOS/IPSL (slides 28 à 38)
5. Projet de groupe ESPRI-IA (slides 39 et 40)
6. Sondage sur votre pratique de l'IA (slide 41)

Concepts

Machine Learning : ensemble de méthodes pour modéliser des données afin de réaliser une tâche définie

Data Mining : fouille de données pour extraire de la connaissances

Big Data : méthodes et technologies d'exploitation de très grands volumes de données au contenu très divers (Volume - Variety - Velocity)



Modèle et Tâche

Qu'est ce qu'un modèle en Machine Learning :

- Fonction mathématique plus ou moins complexe
- Ensemble de règles décisionnelles
- Ne se base pas sur des processus physiques
- **Grand nombre de paramètres (> million pour le deep learning)**

Exemple de tâches d'un modèle (non exhaustif) :

- Régression (ex : calibration d'appareil de mesure, fonction de transfert)
- Classification et reconnaissance de formes (ex : détection d'événements, segmentation)
- Débruitage de signaux (ex : image satellitaires)
- Jouer à un jeu (ex : Alpha Go Zero)

Apprentissage automatique

L'apprentissage automatique consiste à laisser la machine apprendre plutôt que la programmer de façon explicite.

L'entraînement d'un modèle :

- Fixer la valeur des paramètres du modèle selon sa méthode
- À partir d'un jeu de données
- La performance caractérise un modèle (métriques)

On choisit une méthode ML suivant la nature des données et ses objectifs.

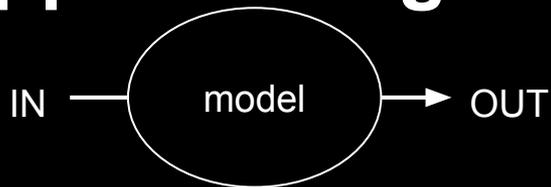
Deux grands types d'apprentissage :

- Apprentissage supervisé
- Apprentissage non supervisé

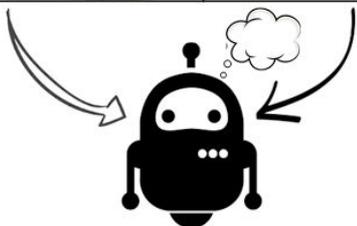
(mais il y a d'autres types ou variantes...)

Apprentissage supervisé

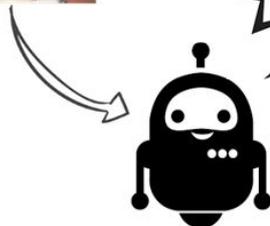
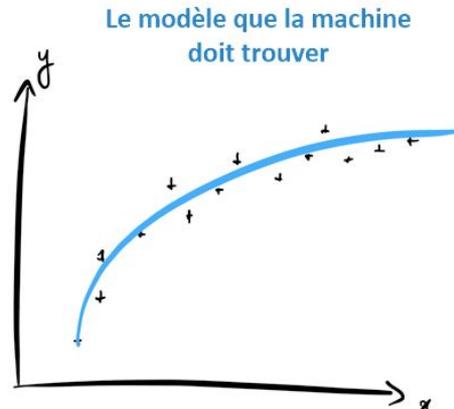
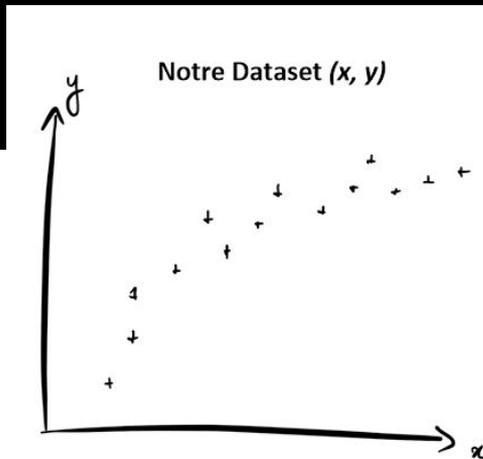
© Guillaume Saint-Cirgue (Machine Learning)



| x | y |
|-----------------------------------------------------------------------------------|---------|
|  | "Chien" |
|  | "Chien" |
|  | "Chat" |
|  | "Chien" |



Apprentissage Supervisé



Utilisation finale

- Données sous forme de paires (entrée/réponse)
- Apprendre à calculer une réponse selon l'entrée
- Prédire la réponse pour une entrée inconnue

Apprentissage non supervisé

Exemple : clustering

- Éléments non annotés
- Apprendre à regrouper les éléments selon leurs similitudes
- Caractériser les groupes trouvés
- Prédire un groupe pour un élément inconnu

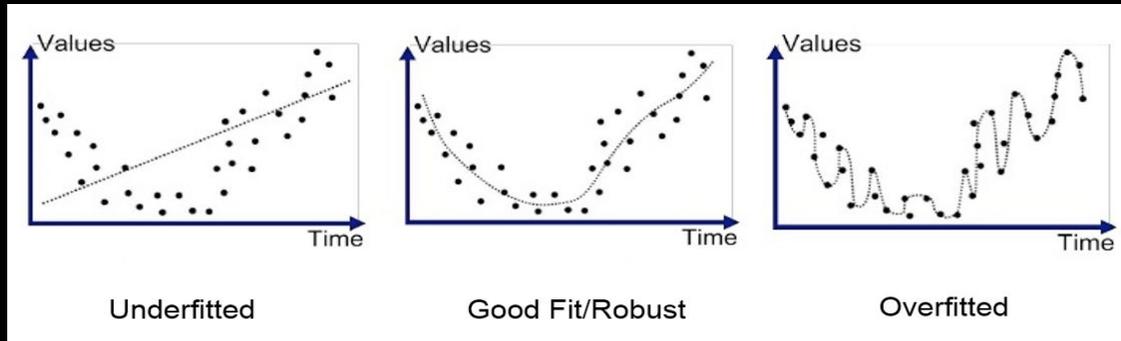


Slides from AS course of P. Gallinari



Principales difficultés

- Choix de la/les méthodes et mesures de performance
- Préparation du jeu de données (vectorisation, échantillonnage, qualité, etc.)
- Problèmes d'entraînement (hyperparamètres, convergence, minima locaux, etc.)
- Généralisation du modèle (under/over fitting)
- Coût d'entraînement du modèle (temps de calculs, mémoire, etc.)
- Coût d'exécution du modèle (cas du streaming de données / temps réel)
- Interprétabilité du modèle entraîné



© Anup Bhande
(Medium)

Neurone formel non bouclé

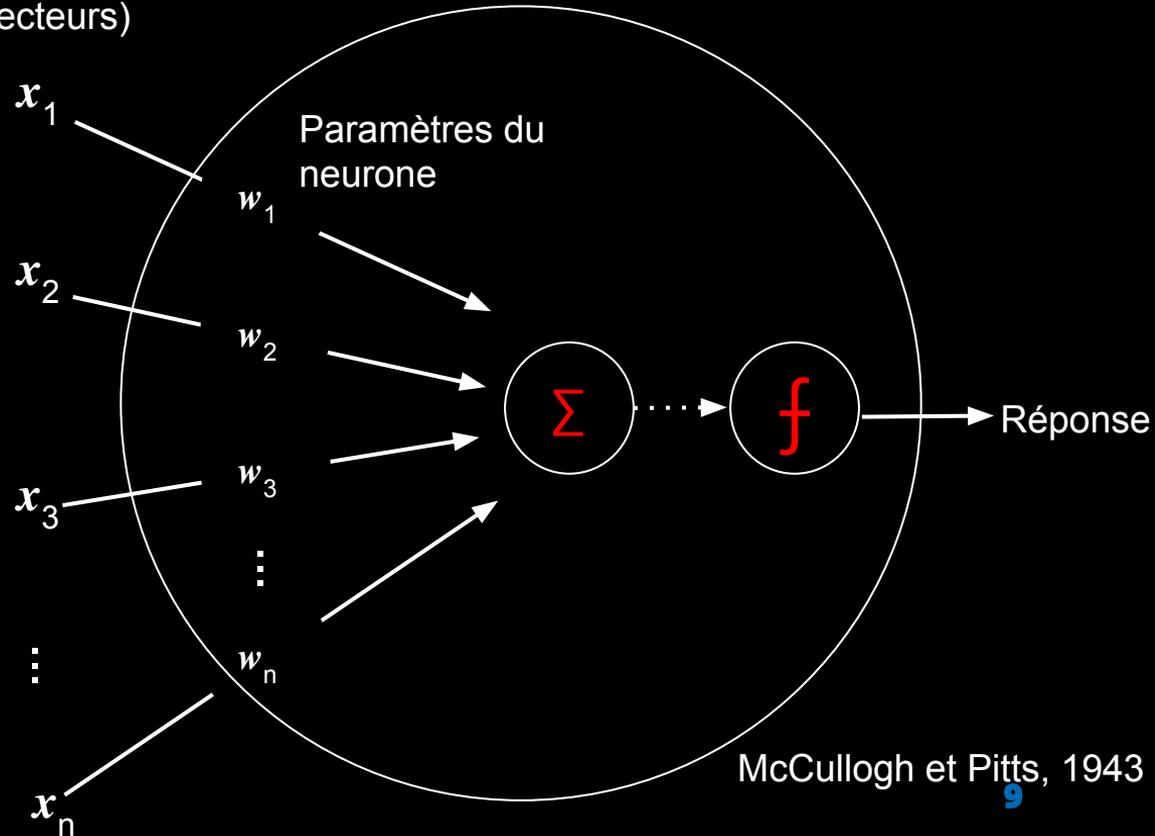
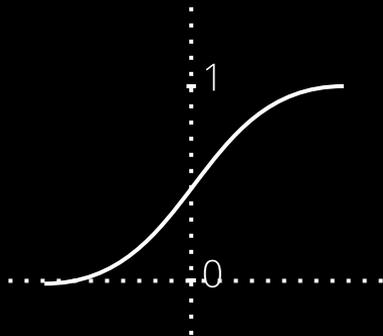
Informations entrantes (sous forme de vecteurs)

Composition de 2 fonctions :

- Fonction linéaire :

$$w_0 + \sum_{i=1}^n w_i x_i$$

- Fonction d'activation (RELU, softplus, sigmoïde, etc.)

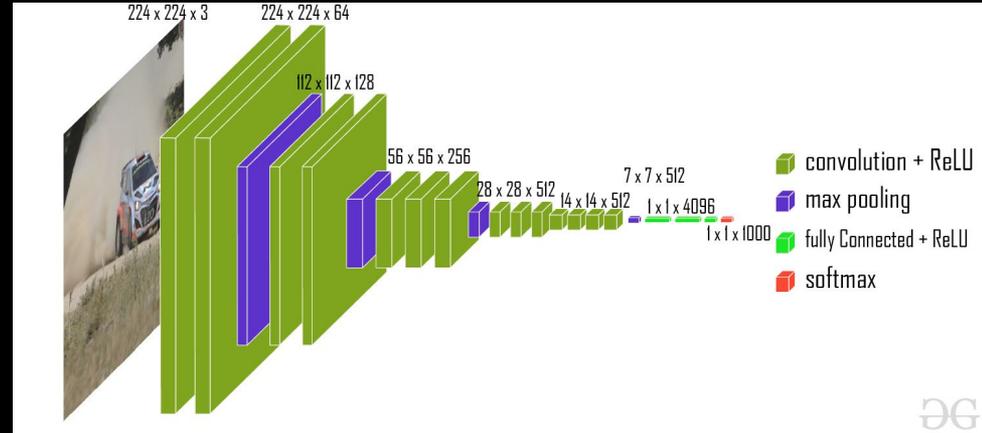


McCulloch et Pitts, 1943

Réseaux de neurones

- Empilements de couches de neurones plus ou moins interconnectés
- Couche de neurones : colonne, grille, volume, hypervolume de neurones
- Heuristiques de construction de réseaux de neurones
- Des plus simples ou plus complexes

© Pawangfg (GeeksForGeeks)



GG

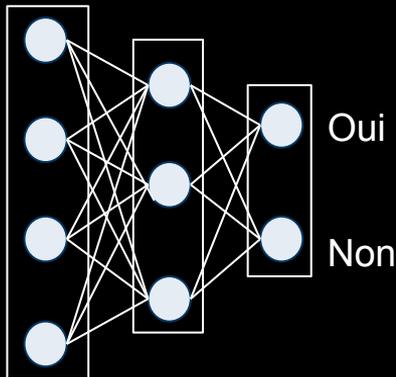
VGG16 : 134 millions de paramètres (ver C)
2 semaines d'entraînement sur 4 x Nvidia TITAN
Karen Simonyan et Andrew Zisserman
Visual Geometry Group, Department of
Engineering Science (University of Oxford)

Température

Pression

Humidité

Pollution



Probabilité
de maladie
?

Deep Learning

Deep Learning : empilement d'un nombre important de couches de neurones

Principales couches de neurones (non exhaustif) :

- Couche de neurones entièrement connectés
- Couche de neurones convolutifs (filtres d'images)
- Couche de neurones récurrents (séries temporelles)
- Couche de neurones récurrents (données à dimension variable)

Quelques architectures de réseaux de neurones (non exhaustif) :

- Réseaux de neurones convolutifs (CNN)
- Auto-encodeurs convolutifs
- Long Short-Term Memory (LSTM)
- Generative Adversarial Network (GAN)

Détection

Objectifs :

Détection et localisation
d'événements locaux
(Cyclone tropical,
Rivière atmosphérique, ...)

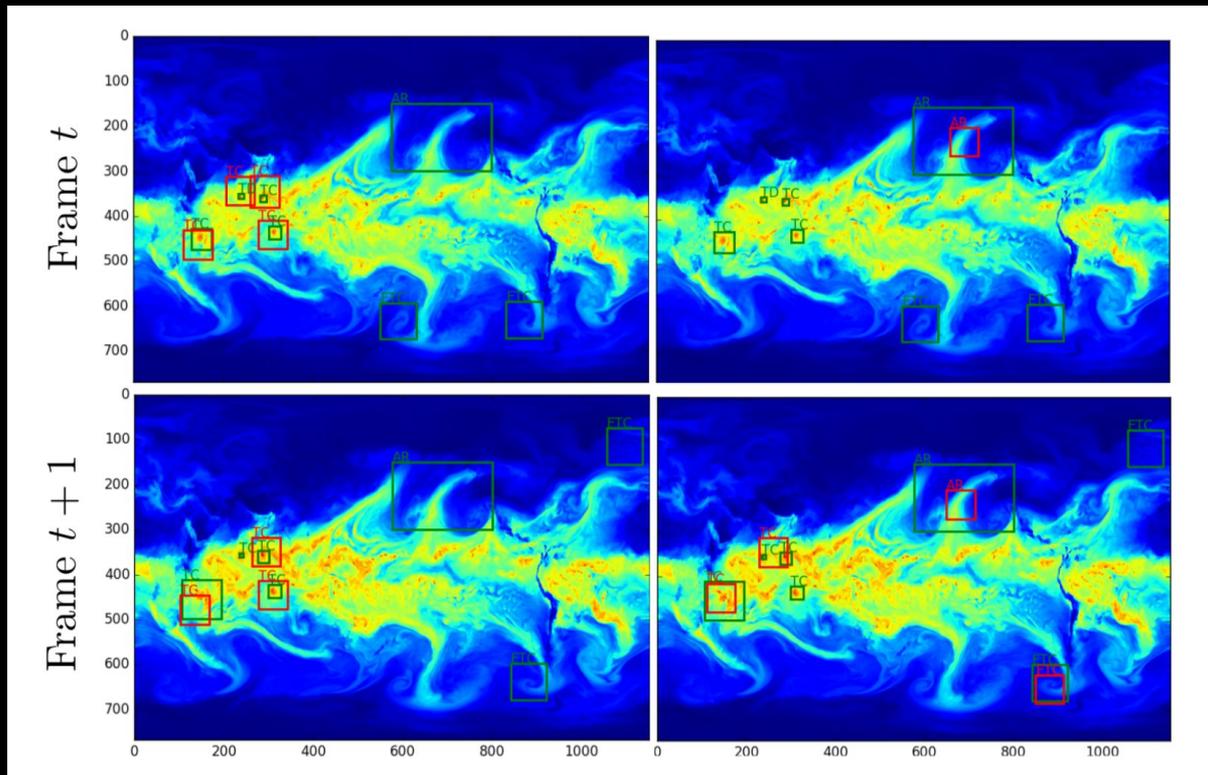
Architecture :

Réseau de neurones
Convolutifs 3D
encoder-decoder
3D = (height, width, time)

Source :

ExtremeWeather : A large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events (2017)

Evan Racah, Christopher Beckham, Tegan Maharaj et al.



Classification

Objectifs :

Classification d'événements météo extrêmes dans les simulations de climat (Cyclone tropical)

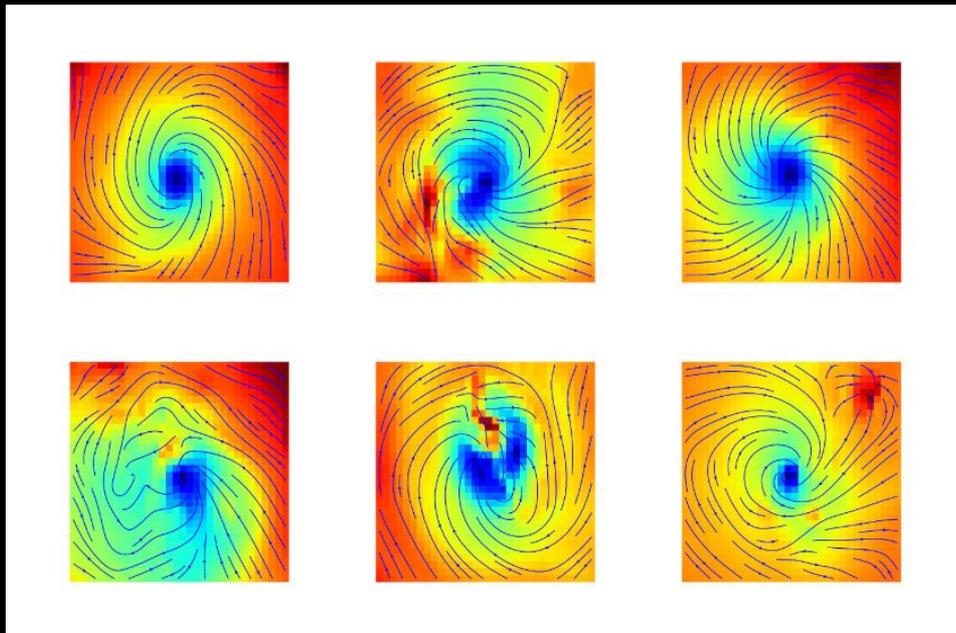
Architecture :

Réseau de neurones convolutifs

Source :

Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets (2016)

Yunjie Liu et al.



Segmentation

Objectifs :

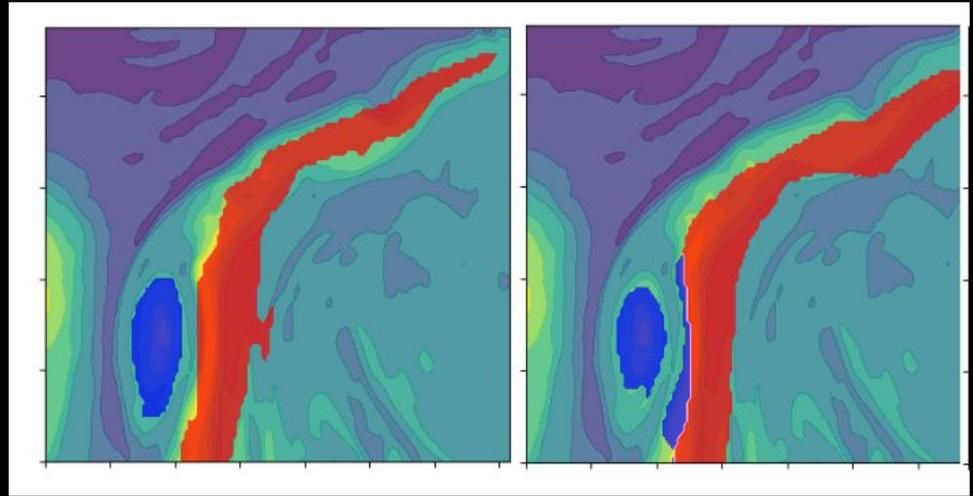
Identifier différent types de zones

Architecture :

Réseau de neurones convolutifs
Réseaux de neurones récurrents

Source :

Segmenting and Tracking Extreme Climate Events using Neural Networks (2017)
Mayur Mudigonda, Sookyung Kim, Ankur Mahesh et al.



Prédiction

Objectifs :

Prédiction de concentration de NO_2 à J+1

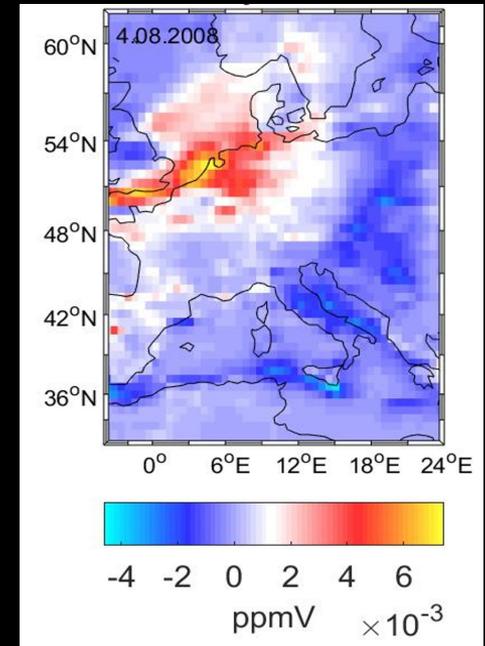
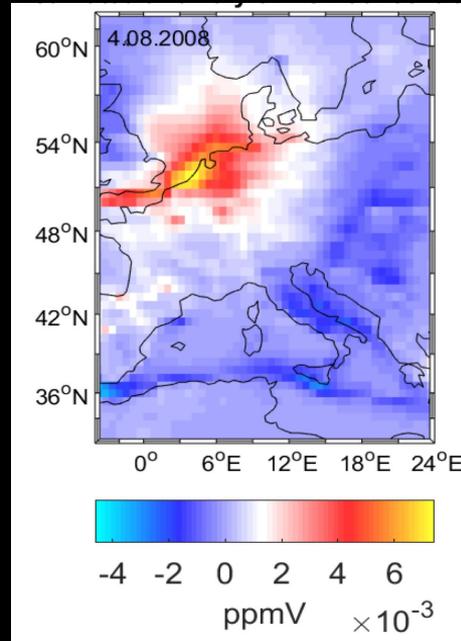
Architecture :

Réseaux de neurones récurrents de séries temporelles

Source :

Estimation of NO_2 and SO_2 concentration changes in Europe from meteorological conditions using a Neural Network (EGU 2020)

Andrey Vlasenko, Volker Matthias, Ulrich Callies et al.



Débruitage

Objectifs :

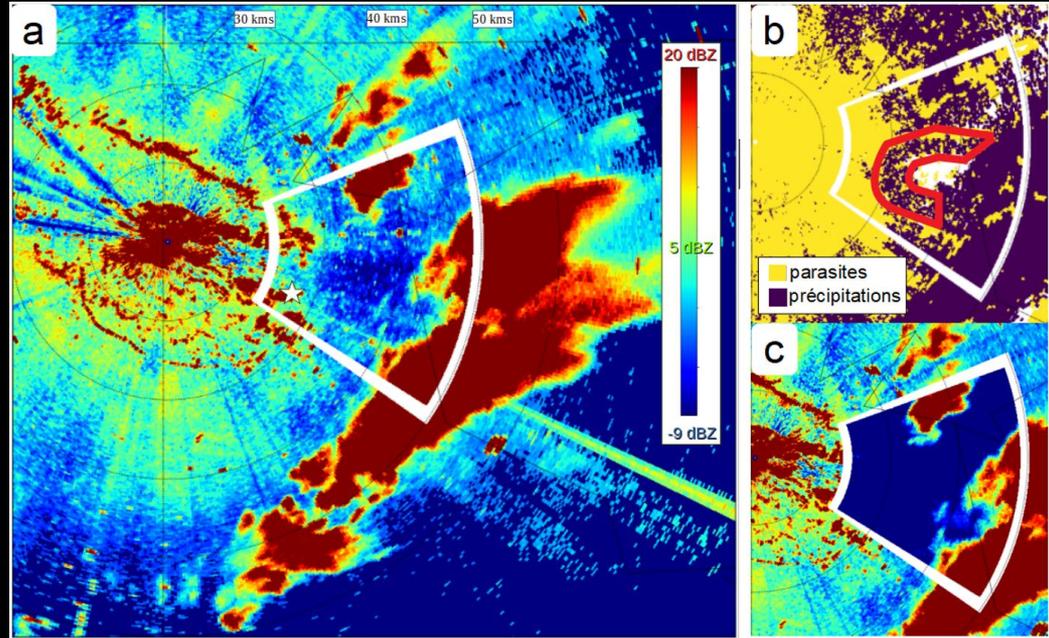
Analyse d'images radars météorologiques pour nettoyage d'échos non pluie

Architecture :

réseau auto-encodeur convolutionnel (U-NET)

Source :

Analyse d'images radars météorologiques
Pierre Lepetit (LATMOS)



Réduction de dimension

Objectifs :

- Représentation des données dans un espace à dimension réduit (espace latent)
- Caractérisation de champs de précipitation
- Comparaison d'observations stations sol et satellites (IMERG)

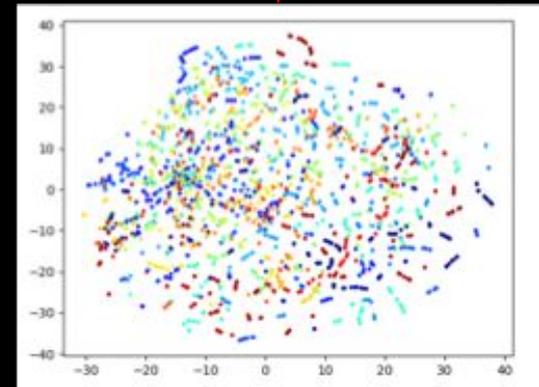
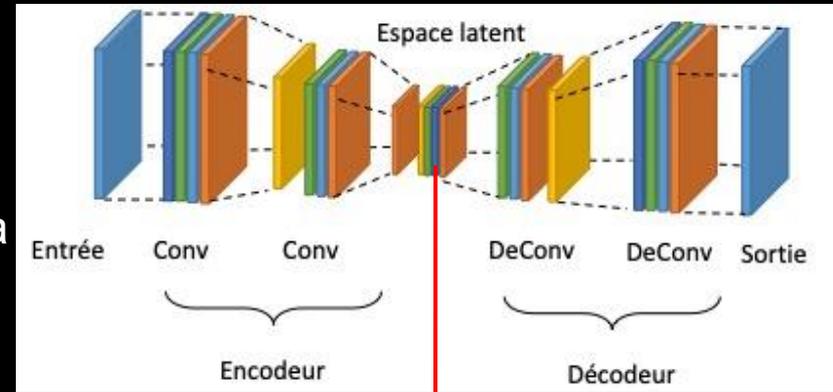
Architecture :

Réseau auto-encodeur convolutionnel

Source :

Analyse de champs de précipitation observés par télédétection spatiale

Laurent Barthes et Cécile Mallet (LATMOS)



Plus de projets

Il existe de plus en plus de sessions Machine Learning dans les congrès sur l'environnement et le climat.

Exemples de présentations :

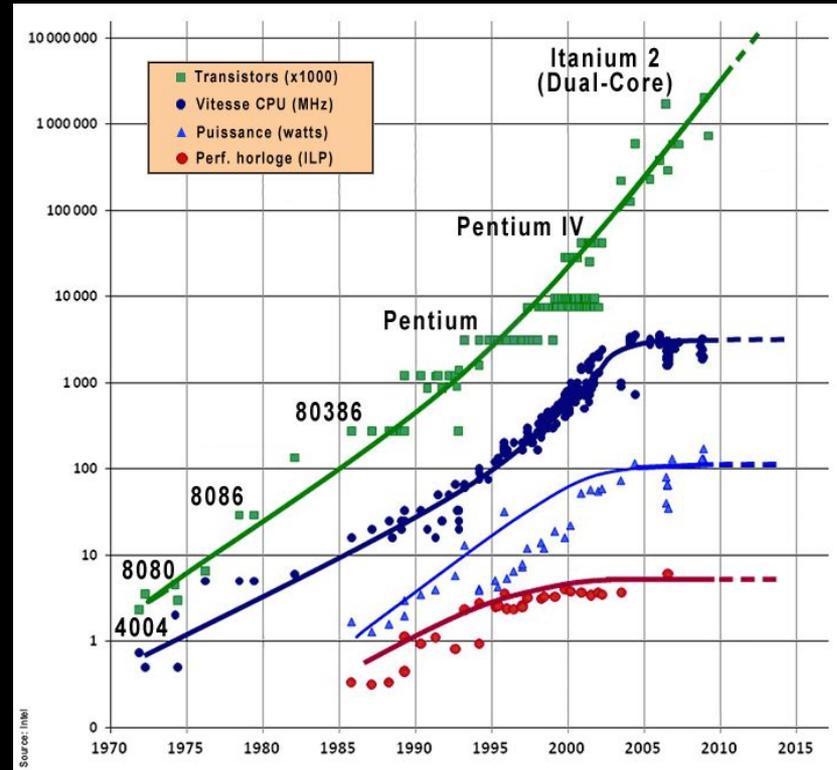
- Climate Informatics 2020 : <https://ci2020.web.ox.ac.uk/>
- EGU 2020 :
<https://meetingorganizer.copernicus.org/EGU2020/sessionprogramme#ITS4>

Les cartes graphiques comme unités de traitements

Historique

Augmentation de la puissance de calcul tous les 18 mois (Loi de Moore):

- La fréquence (finesse de gravure)
- La puissance
- Le nombre de transistors

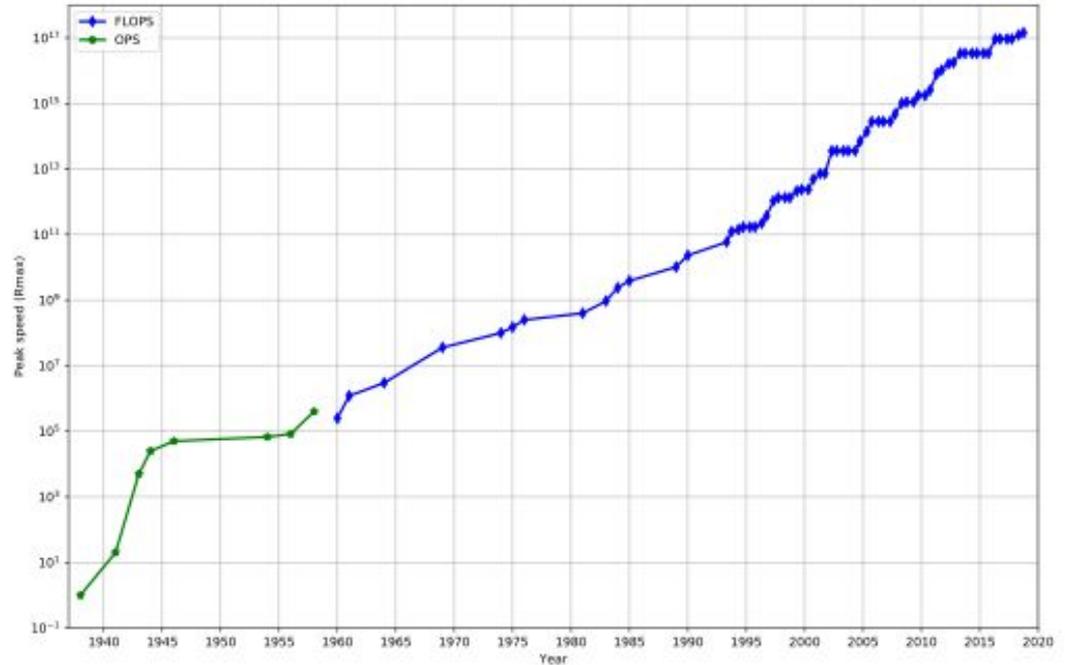


Free lunch is over: A Fundamental Turn
Toward Concurrency in Software
© Herb Sutter (Dr Dobb's Journal)

Historique

Limite des prévisions de Moore prévue en 2022 :

- Augmentation du nombre de cœurs
- Limite de gravure à 3,5 nm

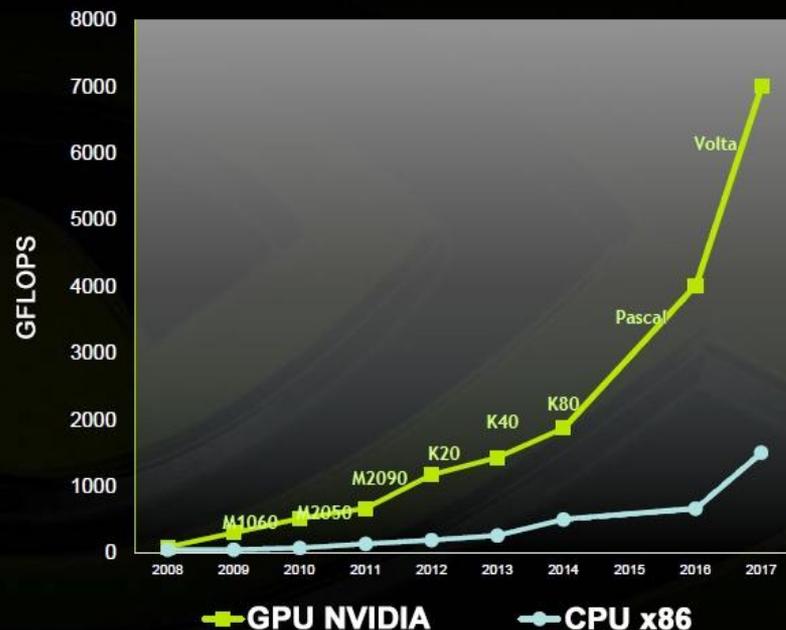


Traitement de données sur GPU

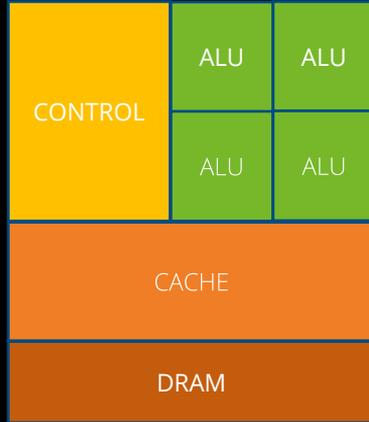
Evolution rapide des GPU (FP64)

- Le dernier CPU 4 TFLOPS
- Le dernier GPU 19,5 TFLOPS

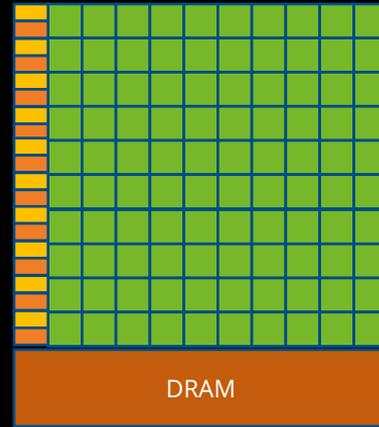
FLOPS en double précision (Pic)



CPU VS GPU

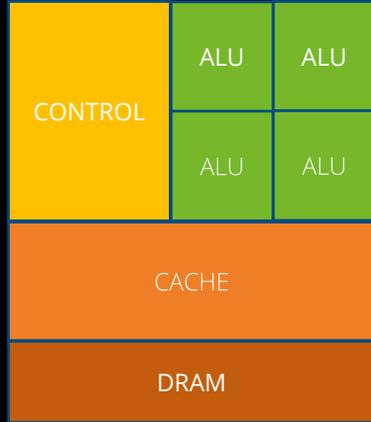


CPU



GPU

CPU VS GPU



CPU

- Peu de cœurs (52 cœurs Q2/2019)
- Haute fréquence d'horloge
- Caches de grandes tailles
- Puissante unité arithmétique et Logique (ALU)
- Contrôle sophistiqué (interprétation et prédiction des instructions)

Chaque processeur exécute son code de manière asynchrone et indépendante

- Optimisé pour les opérations séquentielles

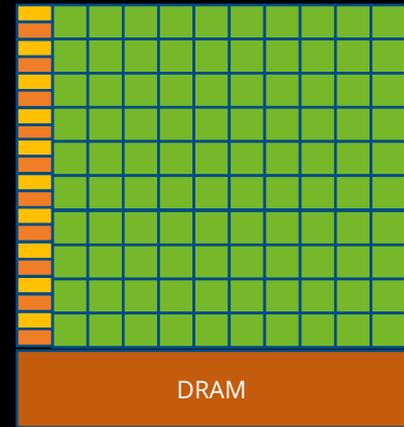
CPU VS GPU

- Unité de calcul très nombreux (~6000 Cœurs)
- Fréquence d'horloge modérée
- Caches de petites tailles
- Unité arithmétique et logique (ALU) faible consommation
- Instructions spécialisées

- Modèle SIMT (Single Instruction, Multiple Thread) :

La même instruction est appliquée à un ensemble des données en 1 cycle d'horloge

- Optimisé pour les parties du code parallélisable



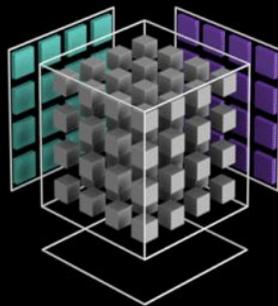
GPU

CPU VS GPU

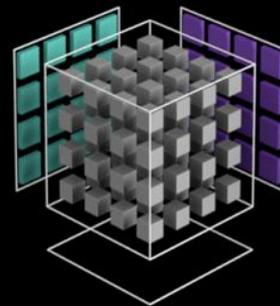
Tensor CORE

- Croisement de matrices en 1 cycle
- Optimum pour les réseaux de neurones à convolutions : 624 TOPS en Tensor INT 8

PASCAL



VOLTA TENSOR CORES



Cluster GPU à l'OVSQ

Objectifs :

- Pouvoir tester les nouvelles méthodes d'analyse
- Voir qu'elles sont les possibilités appliquées à nos activités
- Identifier l'intérêt et la faisabilité
- Pouvoir répondre aux nouveaux besoins d'enseignement
- Evaluer ses besoins pour faire des demandes d'usage du nouveau cluster GPU
JEAN ZAY du GENCI

Regroupement des moyens de :

- OVSQ
- LATMOS
- IPSL
- Projets (LOCEAN, LATMOS)

Architecture du cluster GPU

NVIDIA RTX 2080 TI

- Cœur Nvidia CUDA 4352
- Fréquence 1350 MHz
- Mémoire 11 Go GDDR6
- Bande passante mémoire 616 Go/s
- Pic FP32 : 14,2 TFLOPS (2x Plus puissant que le cluster le plus puissant du monde en 2001)



© NVIDIA

Architecture du cluster GPU

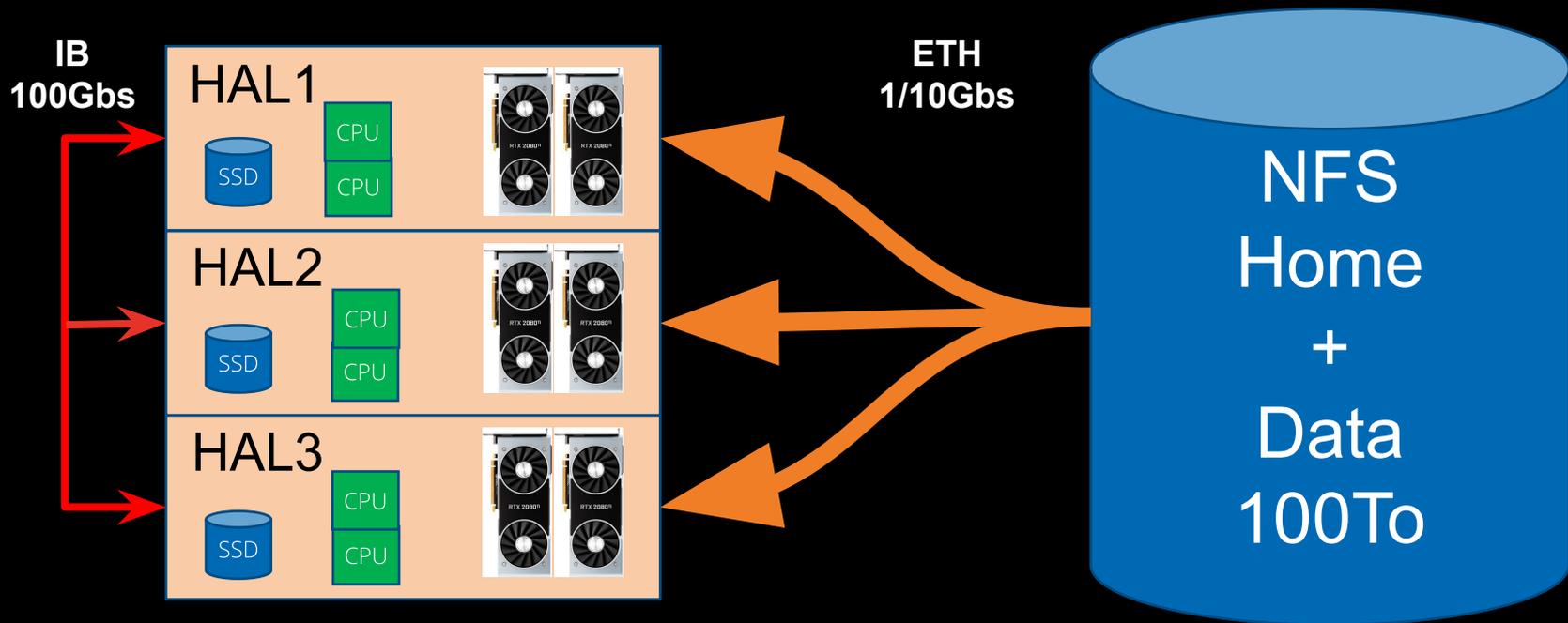
4 Machines :

- 2 x Xeon® 4C/8T 2,6Ghz
- 2 x Nvidia RTX 2080 TI 11Go
- 1 x SSD 4To Lecture intensive



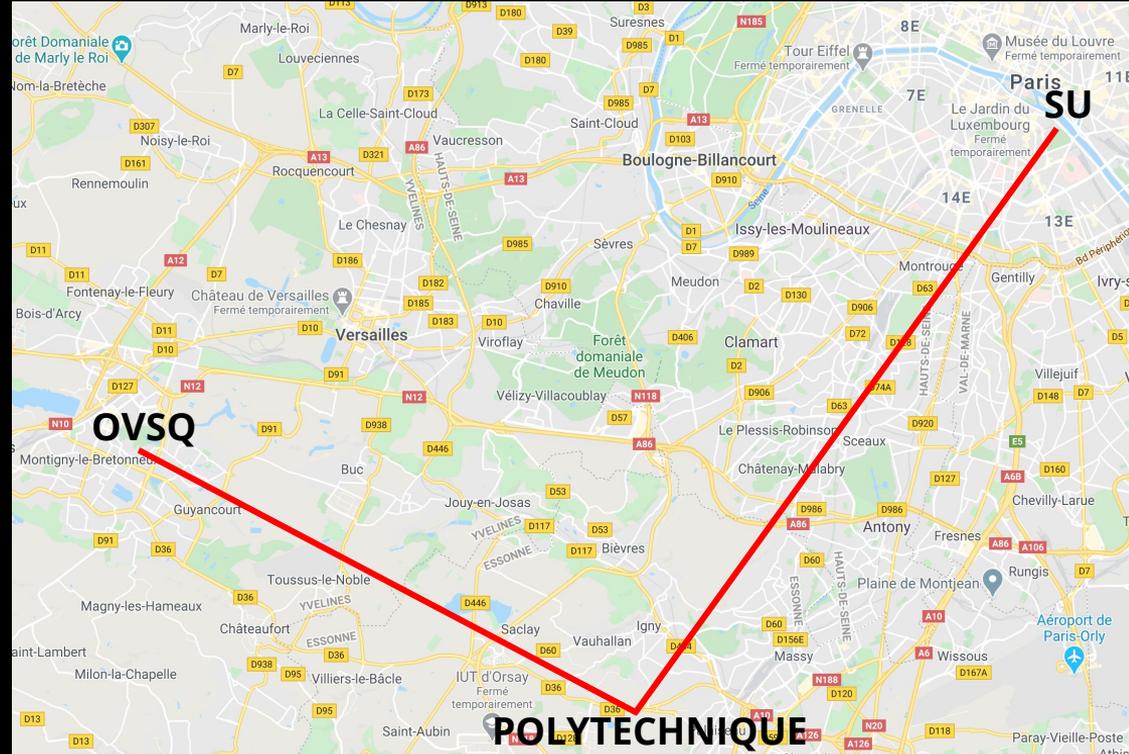
© NVIDIA

Architecture du cluster GPU



Architecture du cluster GPU

- Intégration dans le Mésocentre IPSL
- Interconnexion Très haut débits 10Gbs :
 - OVSQ
 - POLYTECHNIQUE
 - SU



Fonctionnement du cluster GPU

- Machine ouverte à la communauté de l'IPSL
- Demander un compte :
<https://mesocentre.ipsl.fr/ouverture/>
- Se connecter avec sa clef :
`ssh -X hal.obs.uvsq.fr`
- Pour soumettre un job en interactif :
`srun --gres=gpu:1 --x11 --pty bash`

Par défaut, limité à une carte GPU par personne.
Évitez de laisser vos sessions ouvertes.

Fonctionnement du cluster GPU

Les espaces données sont dans :

- Espace disque des mésocentres : **/bdd** et **/data**
- Espace projet NFS : **/net/nfs/** (à demander)
- Espaces disques rapides : **/net/nfs/ssd{x}**

!!Pas de sauvegarde!!

Documentation complète à venir sur le site du mésocentre

Produits disponibles

Liste des produits disponibles :

- les compilateurs
- les bibliothèques

`module avail`

Charger les produits en cohérences
avec la version de CUDA souhaitée :

`module load cuda/10.0`

`module load cudnn/7.5.0cuda10.0`

`module load python/3.7-anaconda2019-10`

```
[delcambre@hal3 ~]$ module avail
----- /usr/share/Modules/modulefiles -----
dot          module-git  module-info  modules      null         use.own
----- /opt/modulefiles/Compilers -----
gnu/4.7.2    gnu/7.2.0    intel/15.0.6.233  nagfor/6.1(default)
----- /opt/modulefiles/Libraries -----
cuda/10.0    hdf5/1.8.19-ifort15    openmpi/1.10.7-gfortran47
cuda/9.0     nccl/2.4.2cuda10.0    openmpi/1.10.7-gfortran72
cuda/9.2     nccl/2.4.2cuda9.2     openmpi/1.10.7-ifort15
cudnn/7.1.4cuda9.0    netcdf4/4.3.3.1-gfortran    openmpi/2.1.2-gfortran
cudnn/7.4.2cuda10.0  netcdf4/4.3.3.1-gfortran47  openmpi/2.1.2-gfortran47
cudnn/7.4.2cuda9.0   netcdf4/4.3.3.1-gfortran72  openmpi/2.1.2-gfortran72
cudnn/7.5.0cuda10.0  netcdf4/4.3.3.1-ifort15     openmpi/2.1.2-ifort15
cudnn/7.5.0cuda9.0   netcdf4/4.5.0-gfortran      openmpi/3.0.0-gfortran
cudnn/7.5.0cuda9.2   netcdf4/4.5.0-gfortran47    openmpi/3.0.0-gfortran47
hdf5/1.8.19-gfortran  netcdf4/4.5.0-gfortran72    openmpi/3.0.0-gfortran72
hdf5/1.8.19-gfortran47  netcdf4/4.5.0-ifort15      openmpi/3.0.0-ifort15
hdf5/1.8.19-gfortran72  openmpi/1.10.7-gfortran
----- /opt/modulefiles/Products -----
cdo/1.9.6    nco/4.6.3
ferret/7.3   opt/modulefiles/Products/matlab/2019a
idl/6.4      python/2.7-anaconda50
idl/8.2      python/3.6-anaconda50
matlab/2019a  scilab/6.0.0
```

Fonctionnement du cluster GPU

- CUDA : plate-forme informatique parallèle et un modèle de programmation développé par NVIDIA pour l'informatique générale sur GPU
- Cudnn : bibliothèque de primitives accélérées par GPU pour les réseaux de neurones profonds, accélérant les routines standards pour tous les frameworks d'apprentissage en profondeur.
- nccl : NVIDIA Collective Communications Library implémente des primitives multi-GPU et multi-nœuds optimisés pour les GPU NVIDIA
- nvcc : Compilateur C en collaboration avec le compilateur C de la machine hôte
- MATLAB 2019a
- Python (environnement à installer) :
 - Pytorch
 - TensorFlow
 - Keras

Fonctionnement du cluster GPU

Voir la disponibilité des cartes GPU avec la commande :

```
nvidia-smi
```

```
[delcambre@hal3 ~]$ nvidia-smi
Tue Jun 11 02:21:11 2019

+-----+
| NVIDIA-SMI 418.56                Driver Version: 418.56          CUDA Version: 10.1         |
+-----+-----+-----+-----+-----+-----+
| GPU   Name           Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf    Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
+-----+-----+-----+-----+-----+-----+
|  0   GeForce RTX 208...   Off   | 00000000:3B:00:0 |              N/A   |
| 41%   31C    P8     12W / 260W | 8803MiB / 10989MiB |      0%   Default  |
+-----+-----+-----+-----+-----+-----+
|  1   GeForce RTX 208...   Off   | 00000000:5E:00:0 |              N/A   |
| 41%   28C    P8     19W / 260W | 247MiB / 10989MiB |      0%   Default  |
+-----+-----+-----+-----+-----+-----+
|  2   GeForce RTX 208...   Off   | 00000000:B1:00:0 |              N/A   |
| 41%   32C    P8     21W / 260W | 247MiB / 10989MiB |      0%   Default  |
+-----+-----+-----+-----+-----+-----+
|  3   GeForce RTX 208...   Off   | 00000000:D9:00:0 |              N/A   |
| 41%   30C    P8     12W / 260W | 1918MiB / 10989MiB |      0%   Default  |
+-----+-----+-----+-----+-----+-----+

+-----+
| Processes:                                     GPU Memory |
|  GPU       PID    Type   Process name                               Usage      |
+-----+-----+-----+-----+-----+-----+
|    0      369237    C     python                                    8793MiB   |
|    1      369237    C     python                                    237MiB   |
|    2      369237    C     python                                    237MiB   |
|    3       79808    C     python                                    1671MiB  |
|    3      369237    C     python                                    237MiB   |
+-----+-----+-----+-----+-----+-----+
```

SPYDER !!!!

VIDER LA MÉMOIRE A LA FIN

Evolutions

Le cluster est déjà fonctionnel mais va évoluer en fonction des besoins.

- Matériel :
 - Etude d'un stockage transverse haute performance
- Logiciel :
 - Ajout de nouveaux produits
 - Etude d'une solution de docker compatible NVIDIA
- Interconnexion :
 - Mur d'image de l'OVSQ pour de la visualisation

Projet de groupe ESPRI-IA

Partage d'expérience en ingénierie de l'IA entre le personnel des laboratoires de la fédération IPSL :

- Groupe ouvert à tous
- Conseils techniques sur les méthodes IA (Machine Learning, réseaux neurones, etc.)
- Conseils sur les technologies (bibliothèques, exécution sur le cluster de GPU, etc.)
- Conseils sur la préparation de données (distribution de calculs, parallélisation, etc.)
- Bonnes pratiques en ingénierie logicielle (intégration continue, git, tests, etc.)
- Veille technologique
- Relais d'événements de formation

Groupe en cours d'élaboration

Complémentarité ESPRI-IA et SAMA IA & Climat

ESPRI-IA :

Ingénierie de l'IA, support, partage d'expérience technique, conseils, veille technologique.

Ingénieurs et chercheurs des labos IPSL

Ensemble de Services Pour la Recherche à l'IPSL

Coordinateur :
guillaume.levavasseur@ipsl.fr

SAMA IA & Climat :

Recherche pluridisciplinaire sur les méthodes de l'IA pour la science du climat et de l'environnement, conférences, séminaires.

Chercheurs des labos de l'IPSL, LIP6, LAMME

Statistiques pour l'Analyse, la Modélisation et l'Assimilation

IPSL - Sorbonne Center for Artificial Intelligence

<https://ai4climate.lip6.fr/>

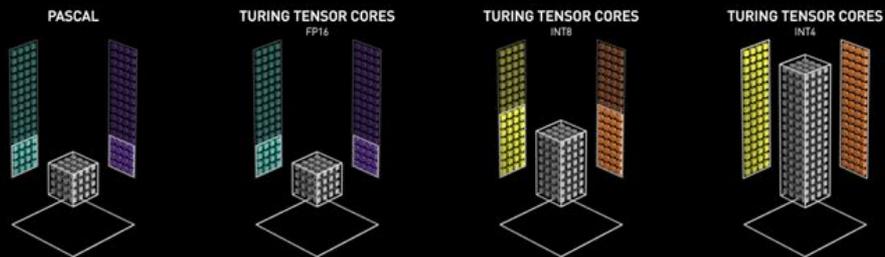
Où en êtes vous?

Petit sondage pour connaître vos pratiques de l'IA ou vos attentes :

- Thème de recherche
- Méthodes d'IA mise en œuvre (kmeans, réseaux de neurones, etc.) et les objectifs (classification, clustering, régression, etc.)
- Langages de programmation et bibliothèques utilisées
- Outillages utilisés (Jupyter, Spyder, Vim, outils cloud, etc.)
- Plateformes de calculs CPU et GPU (Ciclad, HAL, Jean-Zay, AWS, etc.)
- Remarques et/ou attentes

Envoi du lien du sondage par email début juin. N'hésitez pas à partager le lien !

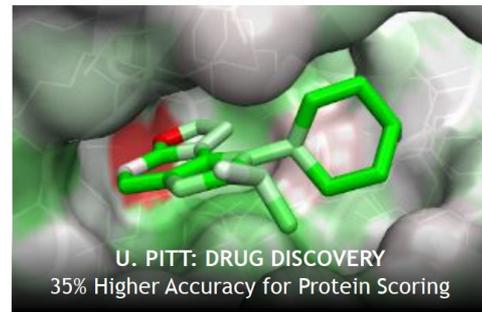
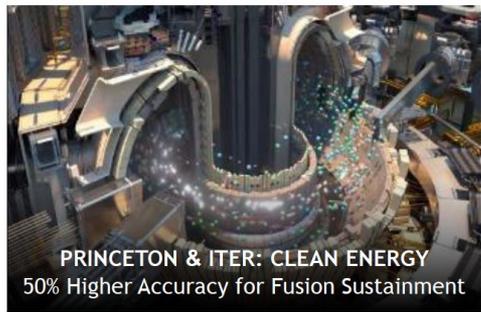
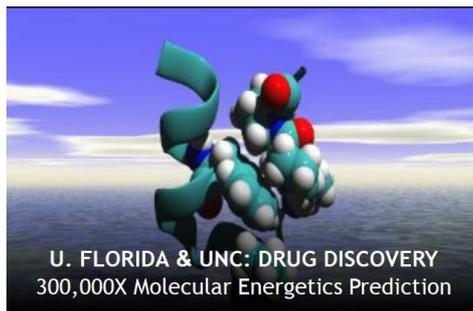
Questions ?



Annexes

Convergence HPC et IA

25% des calculateurs du Top 500 contiennent des GPU



Architecture NVIDIA « Turing »

Architecture GPU :

- Modifications importantes entre chaque génération
- Nécessité d'utiliser les dernières versions logiciels pour exploiter leurs fonctionnalités

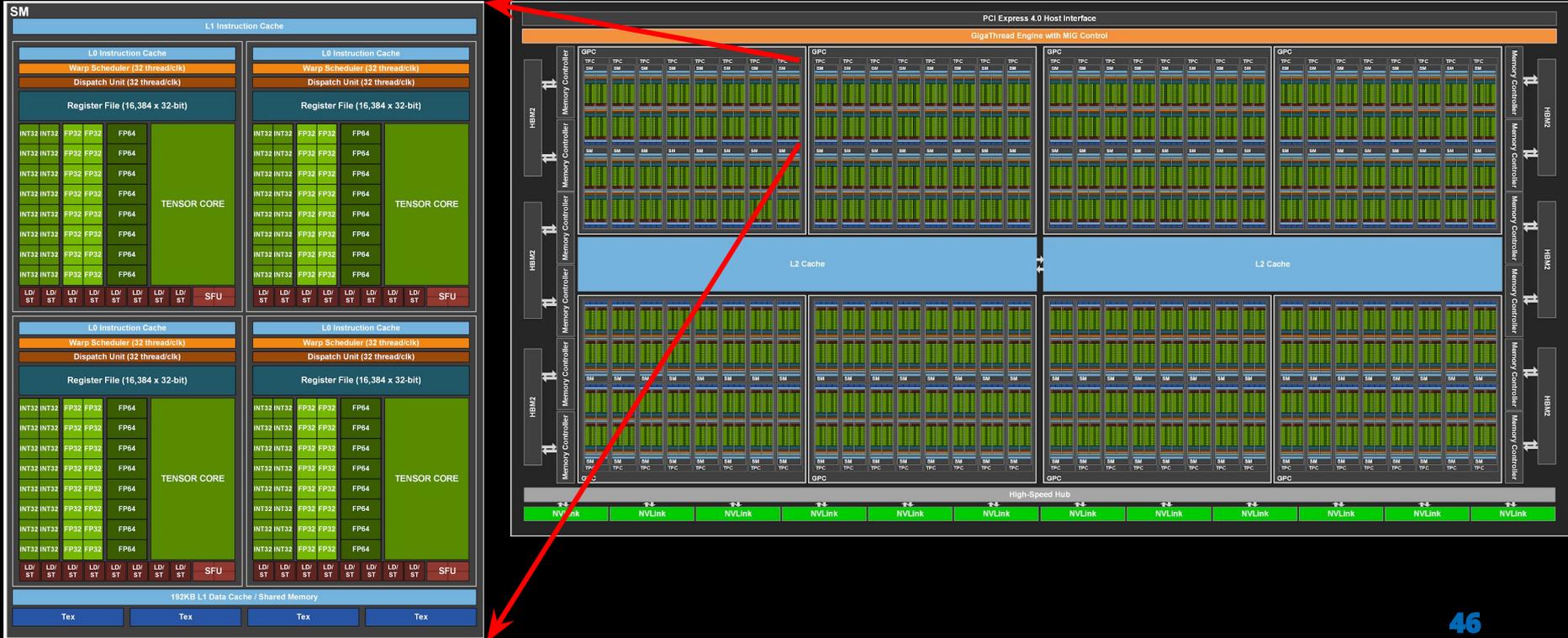


Architecture NVIDIA « Turing »

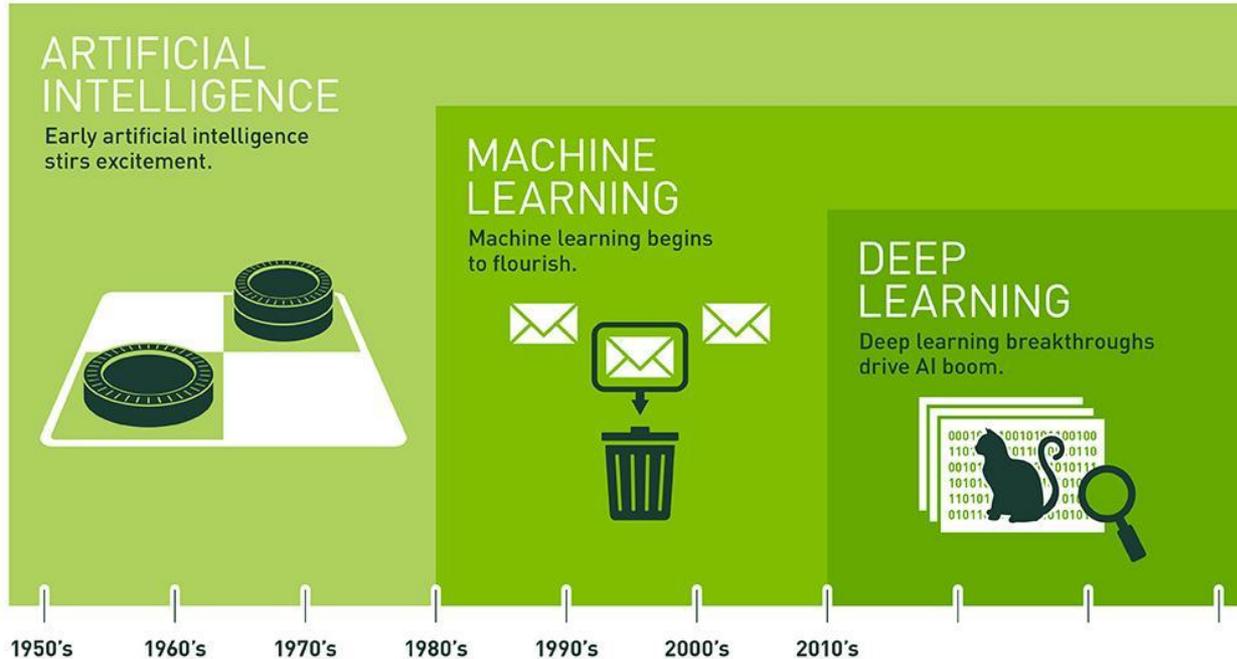
- Changement d'architecture radicale
- Passage d'une unité de traitement
 - utilisable à la fois en INT32 ou en FP32 à des unités de traitement typées : (INT32, FP32, TENSOR CORES)



Dernier GPU NVIDIA A100



Deep Learning



Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

Traitement de données sur GPU

- *2007 CUDA : Compute Unified Device Architecture*

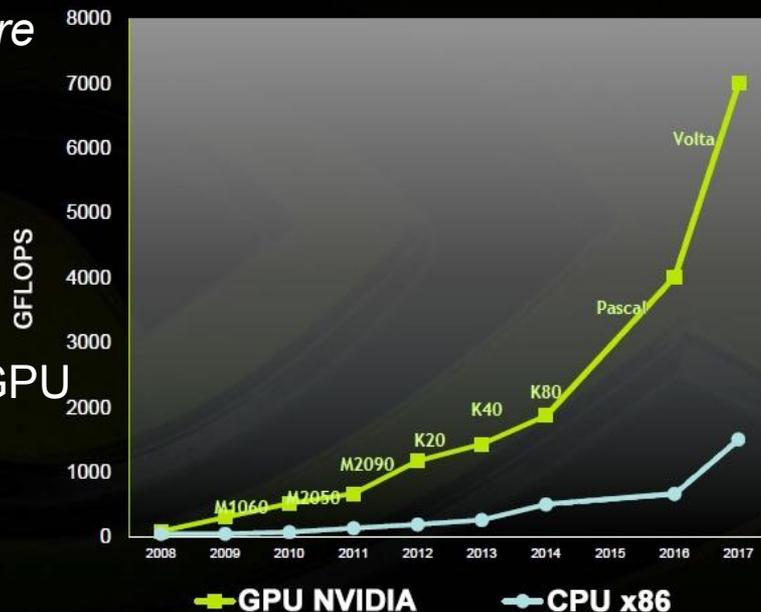
- *Langage spécifique NVIDIA*
- *Optimisation pour le matériel du constructeur*
- *Pas de portabilité sur d'autres équipements*

- *2009 OpenCL : Open Computing Language*

- *Généralisation de calcul Parallèle sur CPU – GPU*
- *Indépendance vis-à-vis du matériel*

Domination du marché par NVIDIA

FLOPS en double précision (Pic)



Deep Learning

- Deep Learning : forme d'apprentissage machine permettant aux ordinateurs d'apprendre de l'expérience et de comprendre le monde en termes d'une hiérarchie de concepts
- Unsupervised learning : est un type d'algorithme d'apprentissage de machines utilisées pour tirer des inférences à partir de fichiers comprenant des données d'entrées sans réactions étiquetées

Deep Learning

DEEP LEARNING IS SWEEPING ACROSS INDUSTRIES

Internet Services



- Image/Video classification
- Speech recognition
- Natural language processing

Medicine



- Cancer cell detection
- Diabetic grading
- Drug discovery

Media & Entertainment



- Video captioning
- Content based search
- Real time translation

Security & Defense



- Face recognition
- Video surveillance
- Cyber security

Autonomous Machines



- Pedestrian detection
- Lane tracking
- Recognize traffic signs

Le jeu un démonstrateur de l'évolution

1996 : Deep Blue gagne contre Garry Kasparov aux échecs

2011 : Watson gagne au jeopardy

2016 : AlphaGo gagne contre Lee Sedol au jeu de go

2017 : Libratus gagne contre des joueurs professionnels dans un tournoi de poker en tête à tête

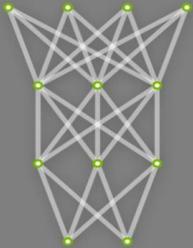
2019 : Pluribus gagne contre 5 joueurs professionnels dans un tournoi de poker à 6

Deep Learning

TRAINING

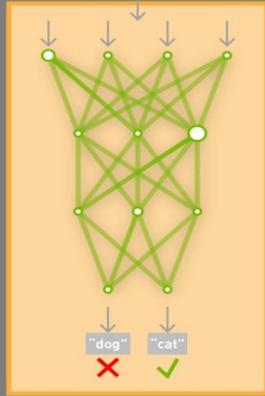
Learning a new capability
from existing data

Untrained
Neural Network
Model

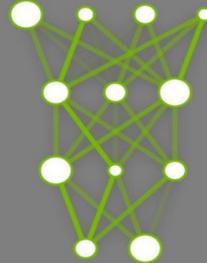


Deep Learning
Framework

TRAINING
DATASET



Trained Model
New Capability



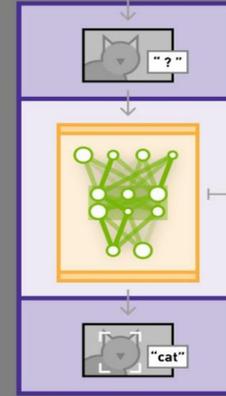
INFERENCE

Applying this capability
to new data

NEW
DATA



App or Service
Featuring Capability

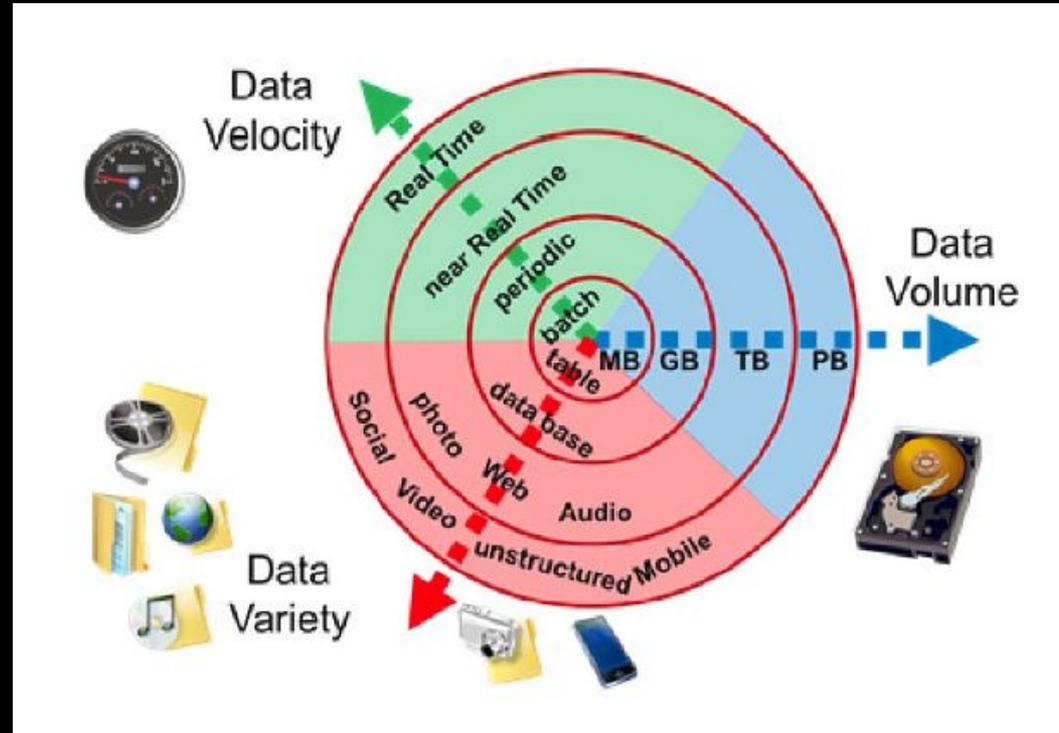


Trained Model
Optimized for
Performance

Big Data

Les 3 V :

- Volume
- Velocity
- Variety



Ender005 (Wikipedia)

Apprentissage par renforcement

- Agent autonome
- Environnement
- Ensemble d'actions
- Objectifs/Récompenses

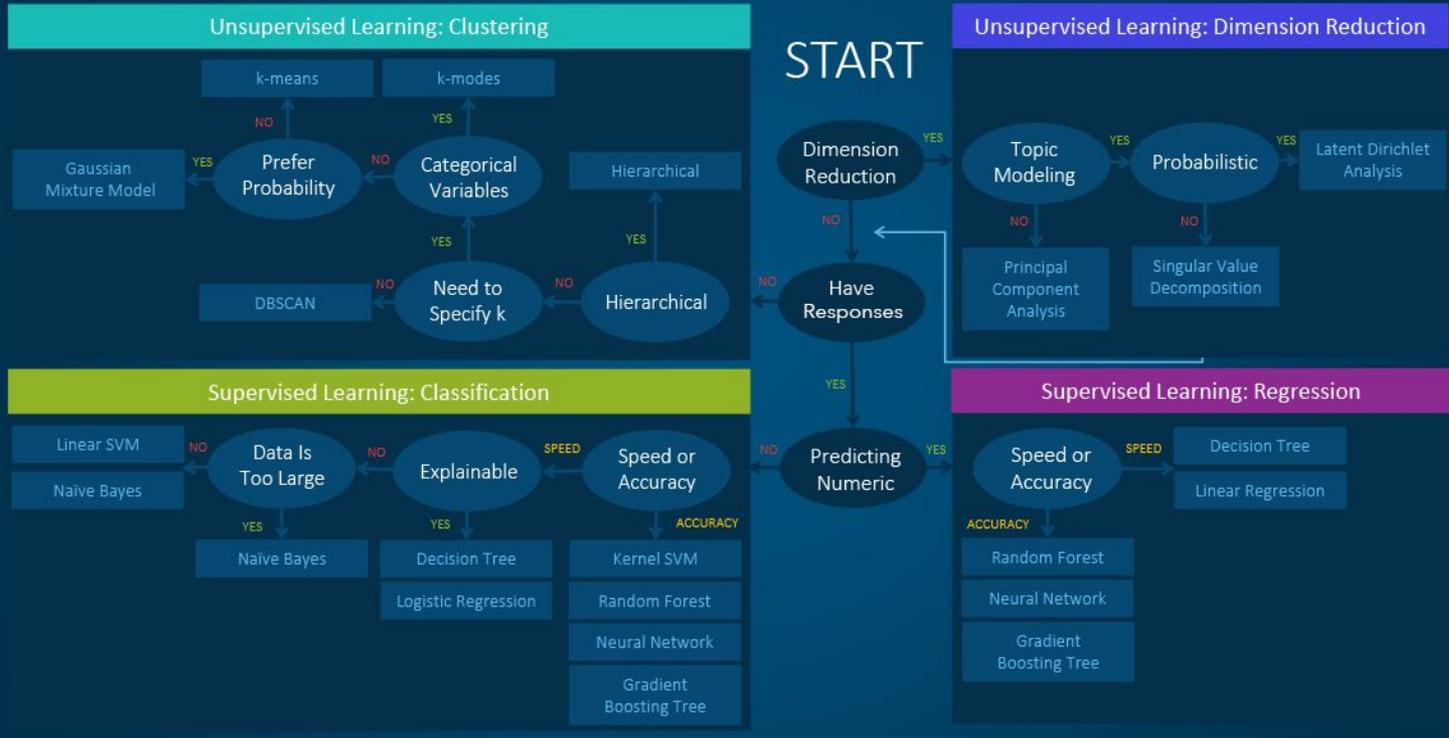


Hunter Heidenreich (Medium)

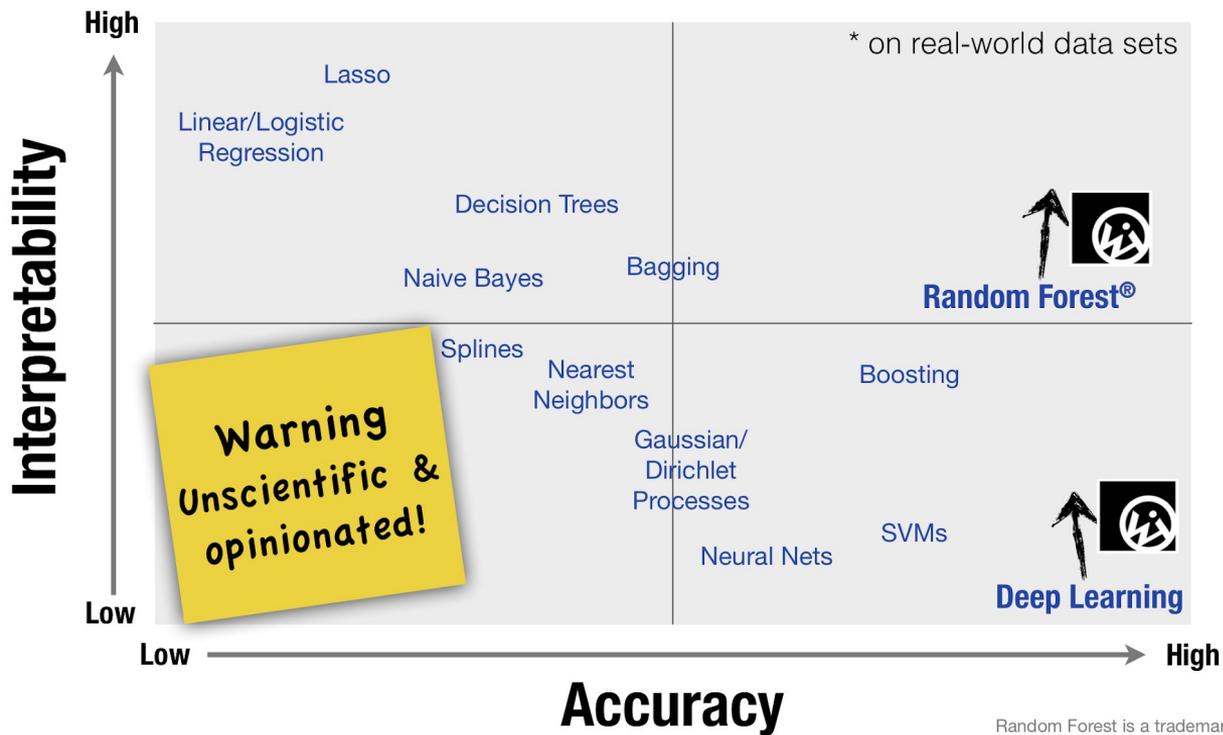
Machine Learning Cheat Sheet

Hui Li (SAS Blogs)

Machine Learning Algorithms Cheat Sheet

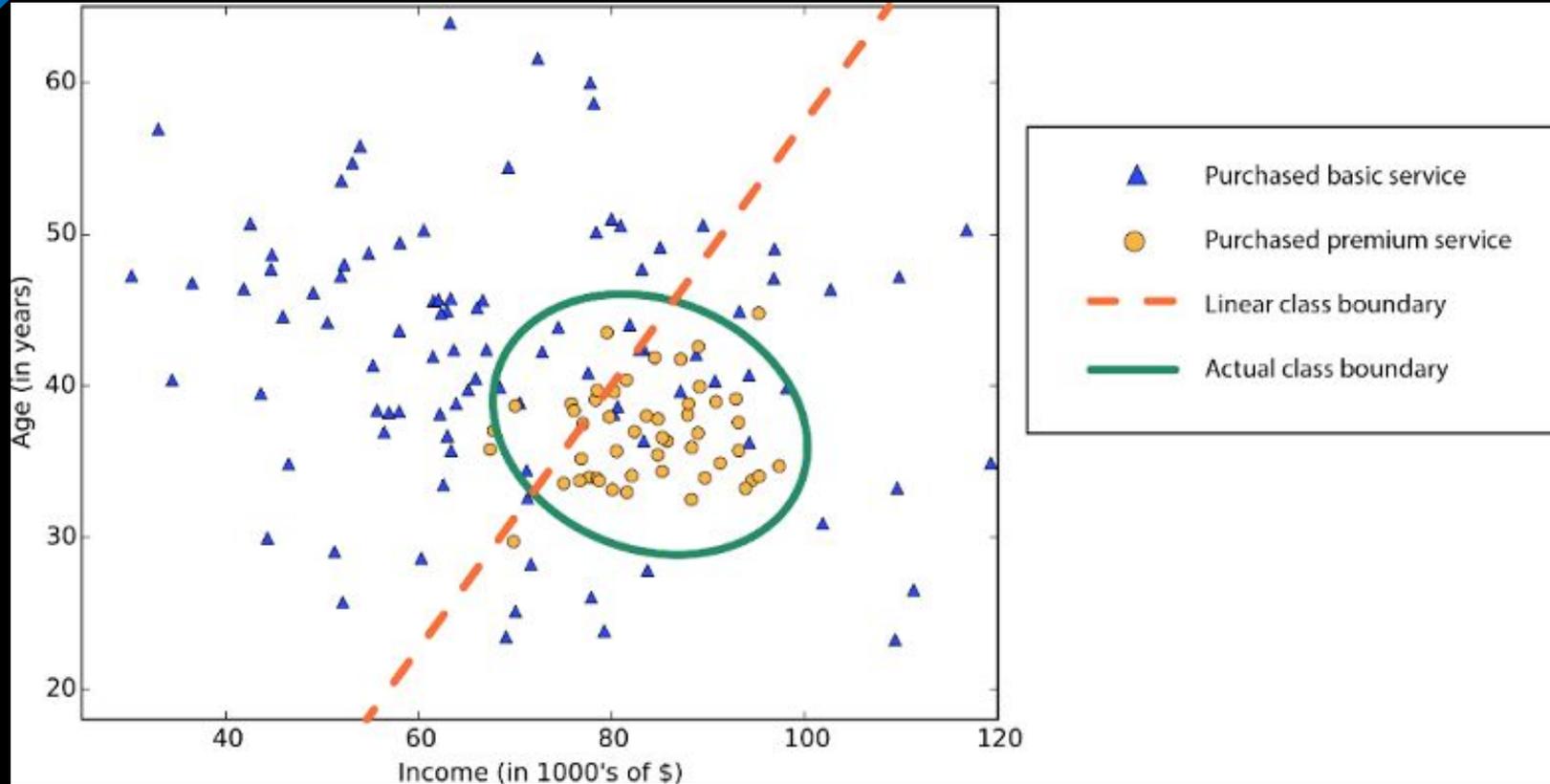


ML Algorithmic Trade-Off

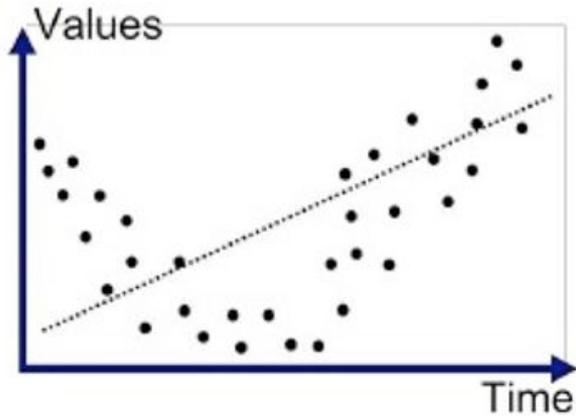


Linéarité

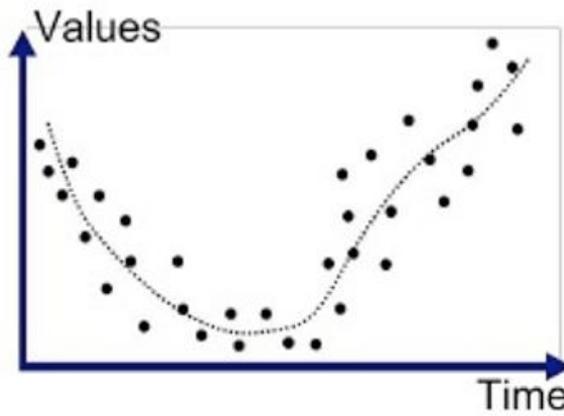
Sélectionner des algorithmes
Azure Machine Learning
Microsoft Build



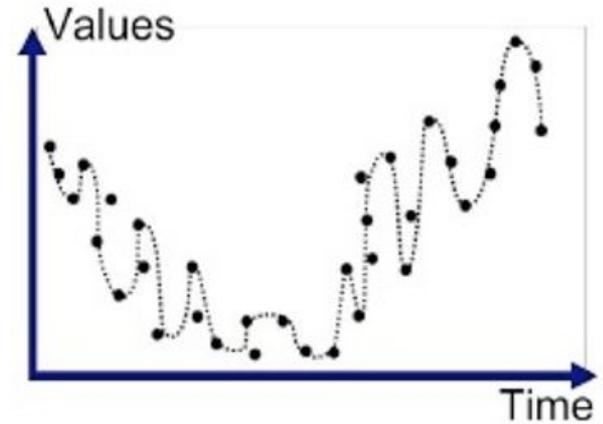
Sur & Sous Apprentissage



Underfitted



Good Fit/Robust

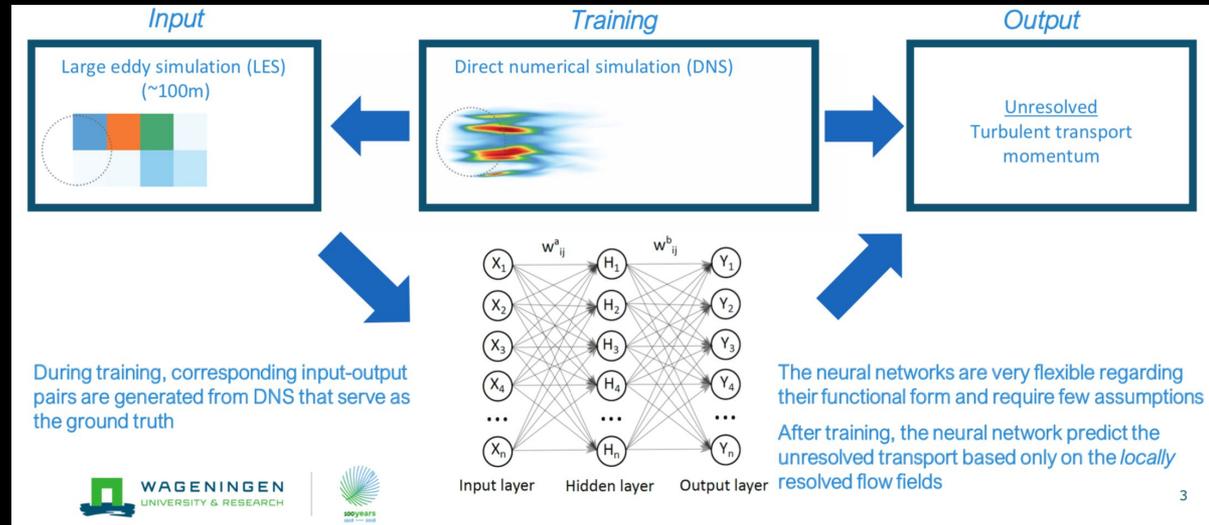


Overfitted

Model downscaling

Objectifs :

- Modélisation de l'humidité au dessus de champs irrigués dans le désert (turbulences atmosphériques).
- Etude du remplacement des modèles à haute résolution



Architecture :

Perceptron multicouches

Source :

LES subgrid modelling using neural networks (EGU 2020)

Stoffer et al



Lepetit et al. : Using Deep Learning for Restoration of Precipitation Echoes in Radar Data, IEEE Transaction geoscience and remote sensing, 2020 (en revision)