



IA, Machine Learning, Deep Learning

Démystification, état des lieux et retours sur une passion professionnelle.



Présentation

Présentation



Présentation - Formation



Présentation - Formation



Présentation – Enseignement



Présentation – Enseignement



Présentation – Enseignement



OPENCLASSROOMS

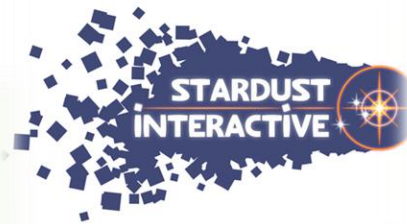
Présentation – Parcours Professionnel

The Thales logo is displayed in a dark blue, bold, sans-serif font. The letter 'A' is stylized with a small teal dot above it. The logo is centered within a white rectangular box that has a subtle drop shadow.

THALES

Présentation – Parcours Professionnel

THALES



Présentation – Parcours Professionnel

THALES



TEKNEO

Présentation – Parcours Professionnel



THALES

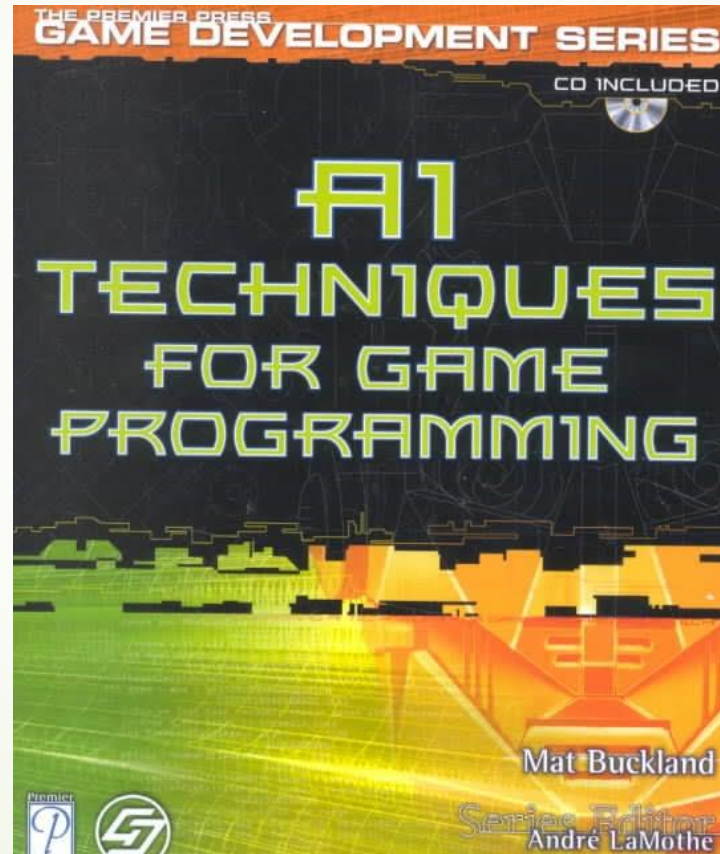


TEKNEO

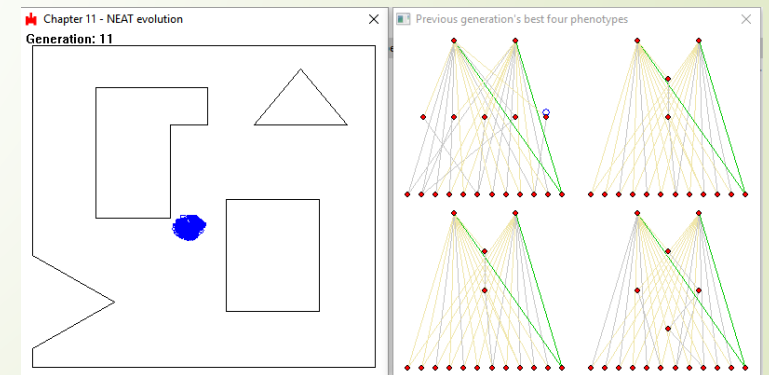
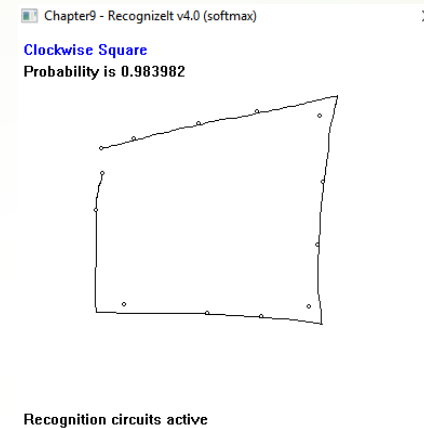
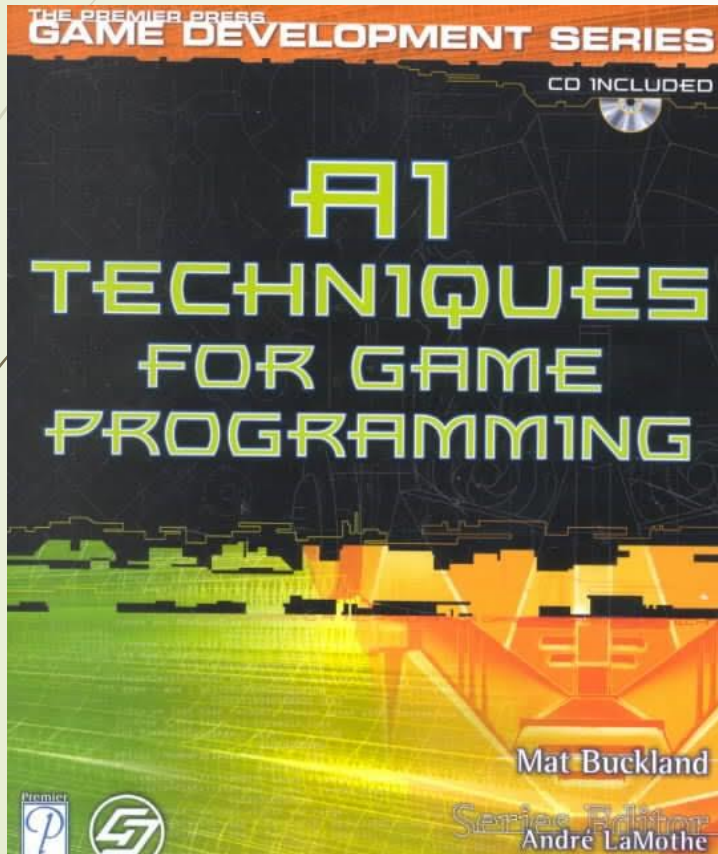


is cool
ENTERTAINMENT

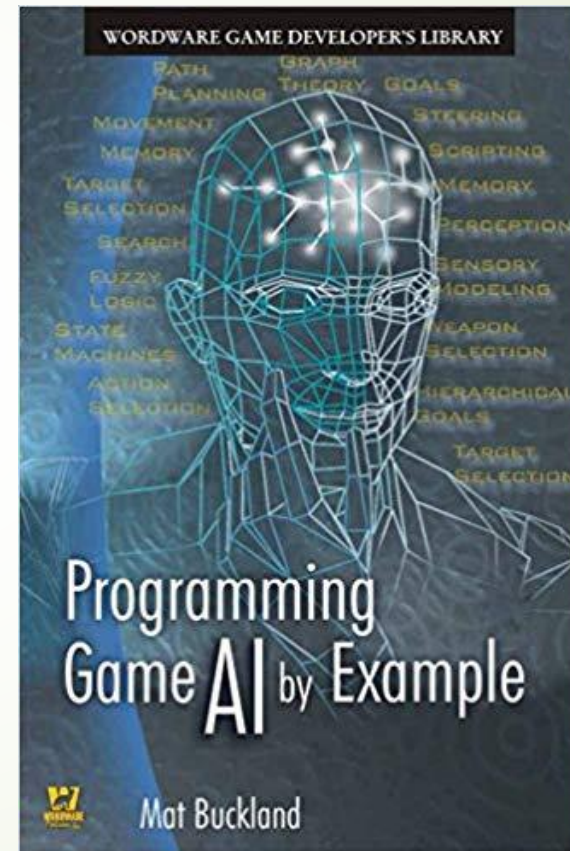
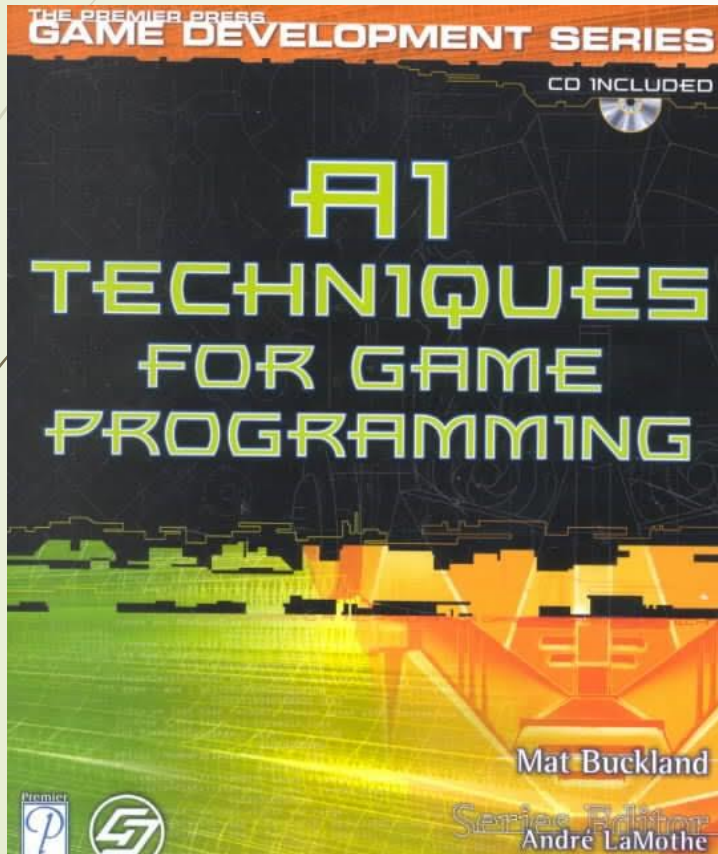
Mes débuts dans la discipline



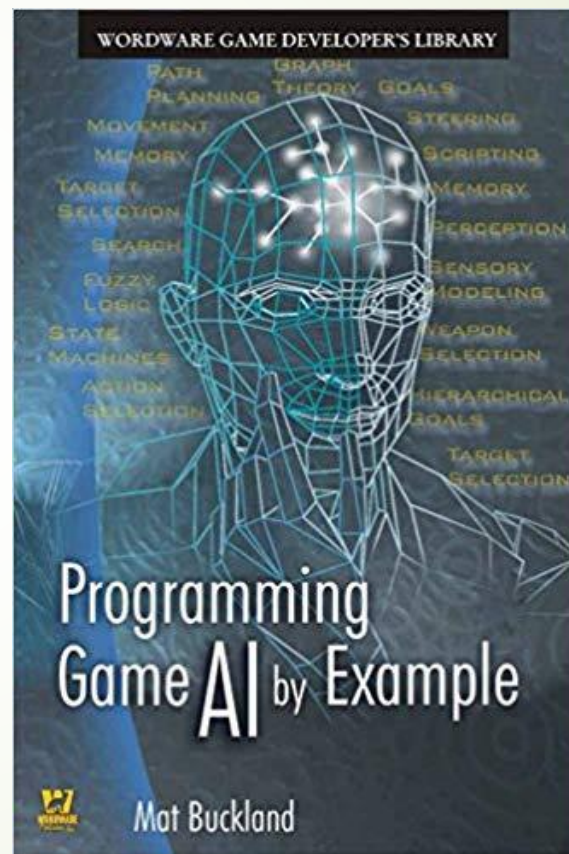
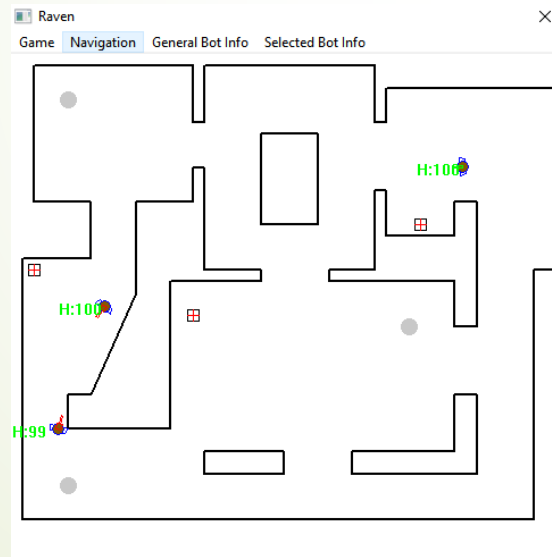
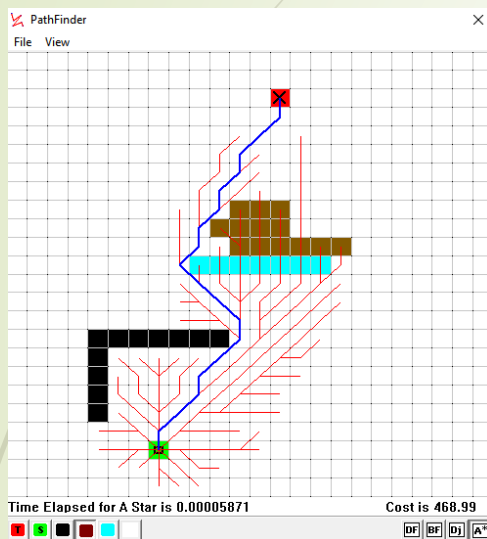
Mes débuts dans la discipline



Mes débuts dans la discipline



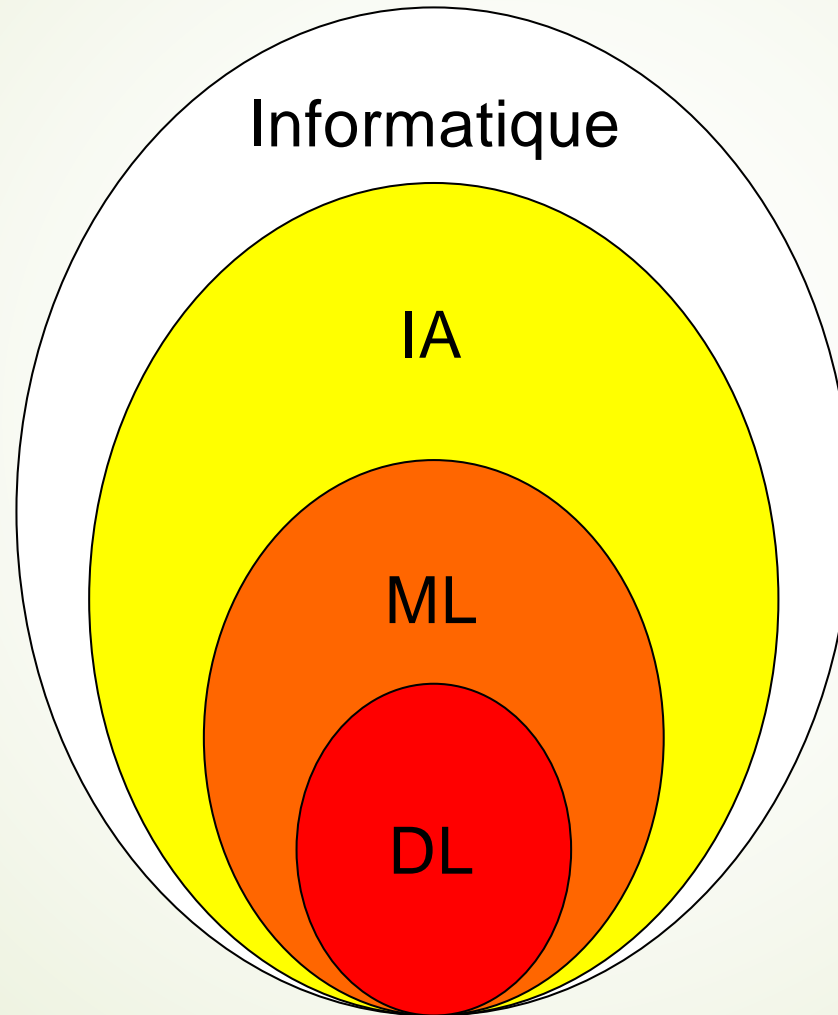
Mes débuts dans la discipline



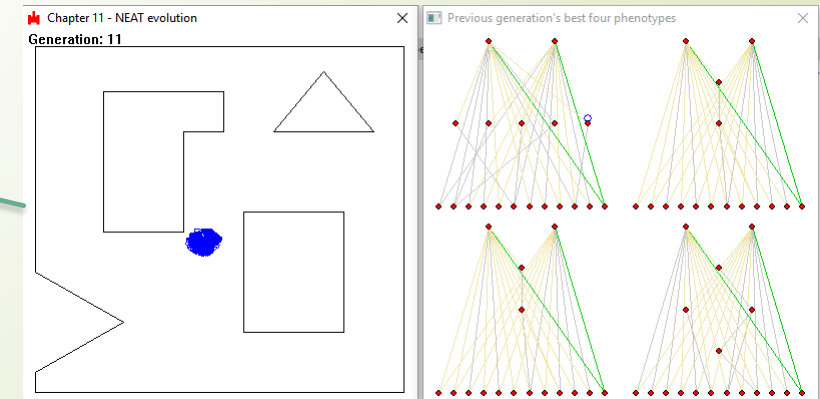
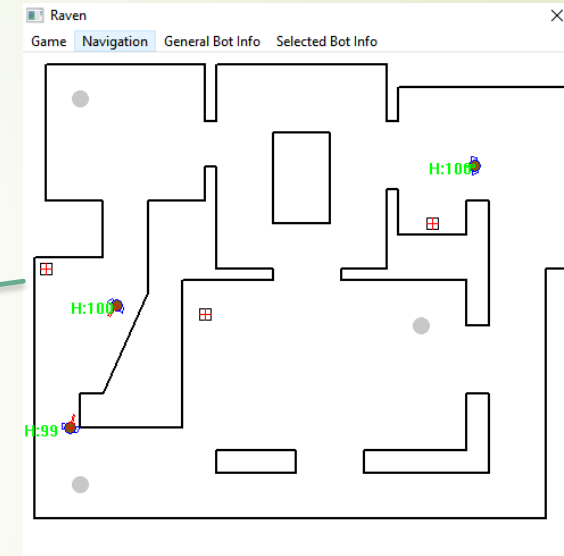
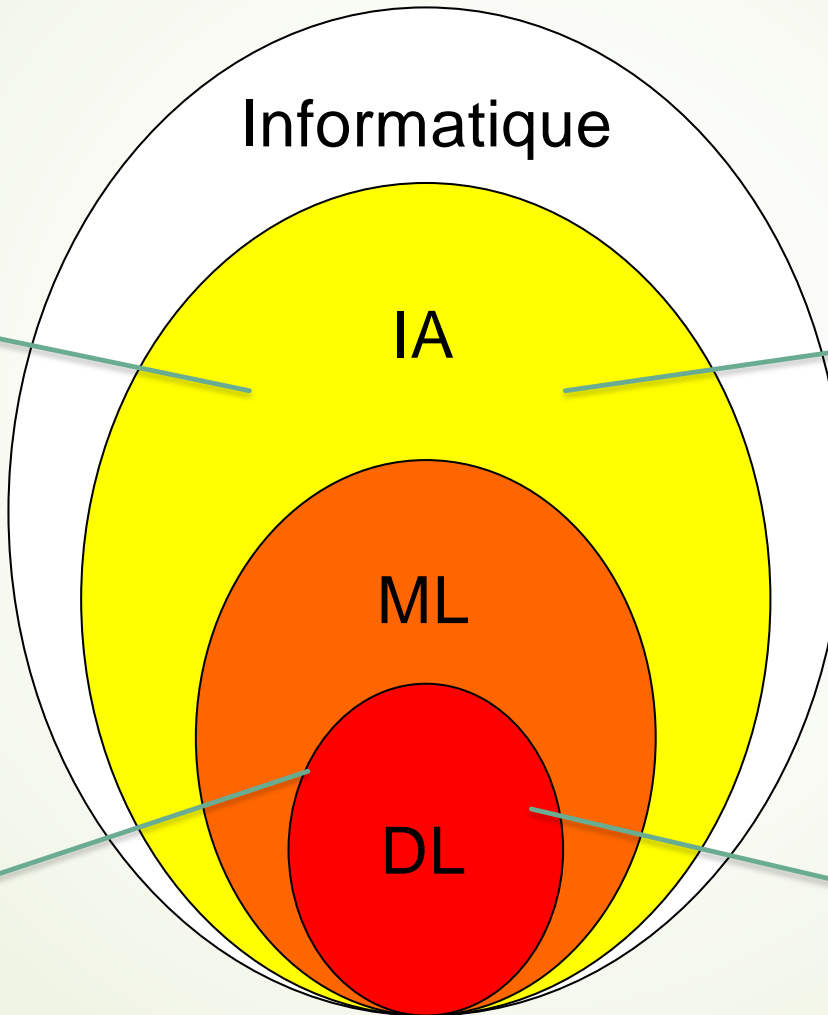
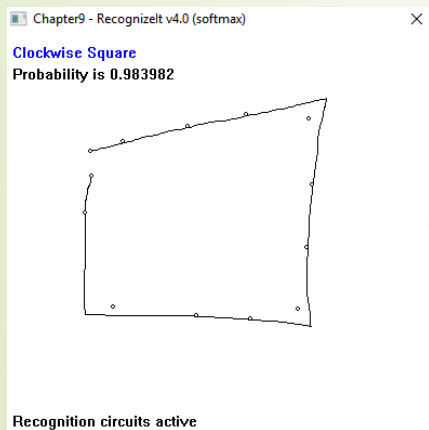
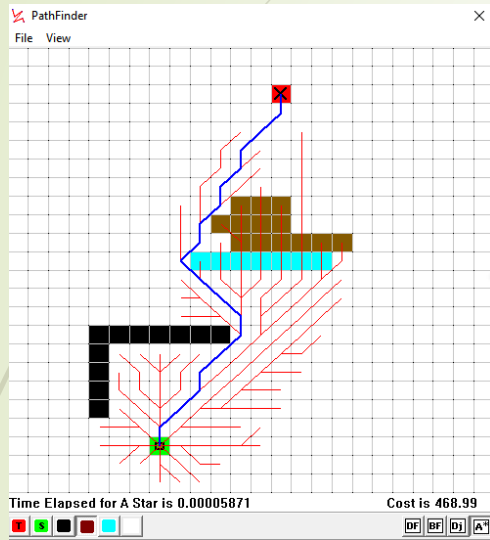


Terminologies

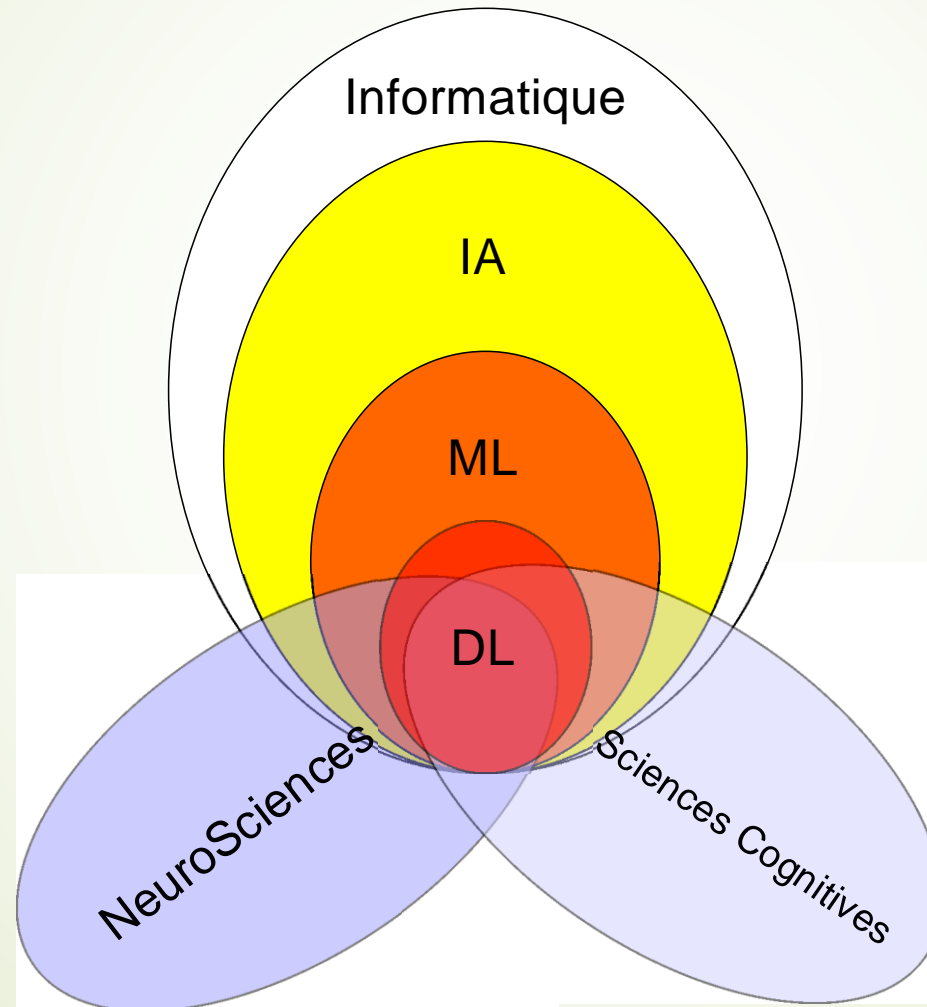
Distinctions entre les termes



Distinctions entre les termes



Distinctions entre les termes





Intelligence Artificielle

Distinctions entre les termes



Intelligence Artificielle (M. Minsky)

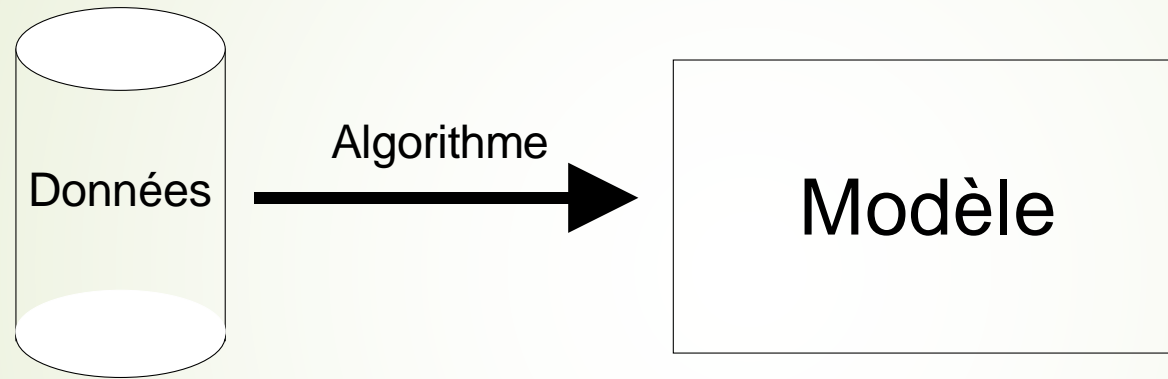
C'est "la construction de programmes informatiques qui s'adonnent à des tâches qui sont, pour l'instant, accomplies de façon plus satisfaisante par des êtres humains car elles demandent des processus mentaux de haut niveau tels que : l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critique"



Machine Learning

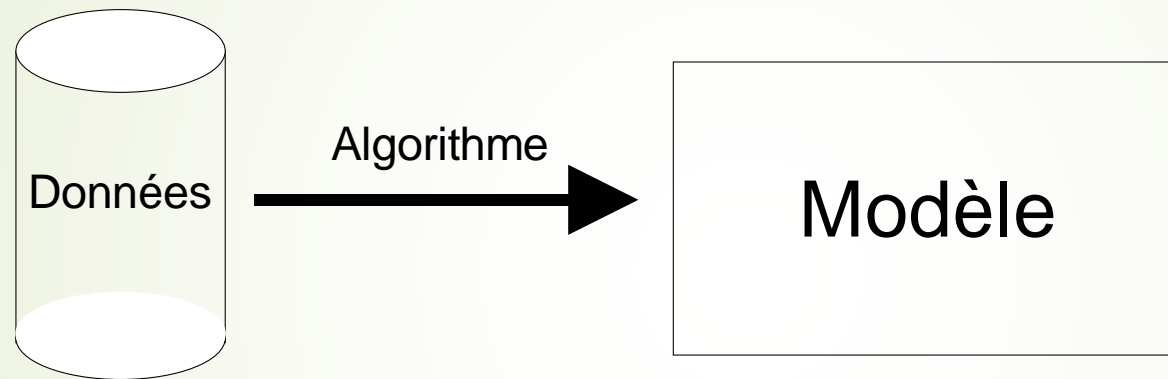
Machine Learning

Apprentissage / Training

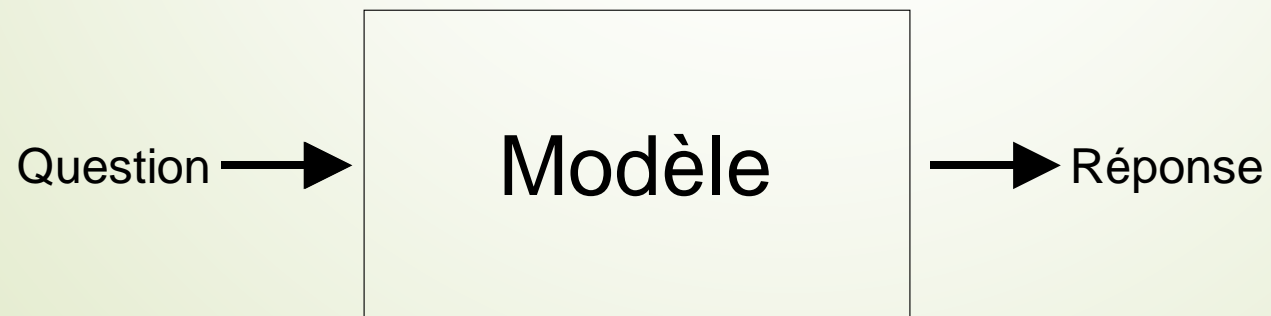


Machine Learning

Apprentissage / Training



Utilisation / Inference





Machine Learning

A quoi ressemble un modèle ?





Machine Learning

A quoi ressemble un modèle ?

Pour être général il s'agit d'une fonction





Machine Learning

A quoi ressemble un modèle ?

Pour être général il s'agit d'une fonction

Pour être TRES général il s'agit d'une distribution de probabilités

Machine Learning

Exemple :

Problème jouet : trouver une fonction (un programme ?) qui soit capable de prédire la taille d'un humain en fonction de son âge...

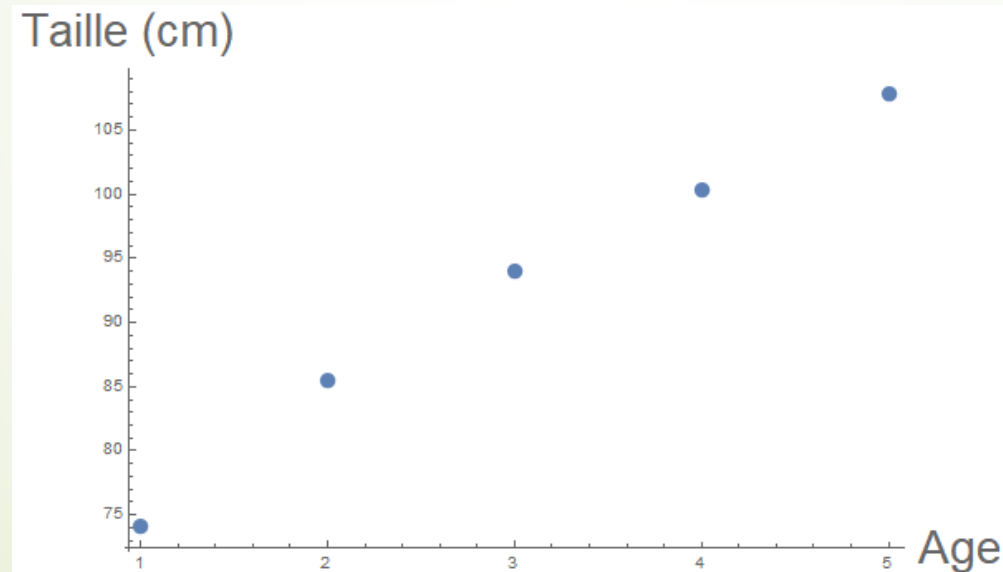


Machine Learning

Exemple :

Problème jouet : trouver une fonction (un programme ?) qui soit capable de prédire la taille d'un humain en fonction de son âge...

Données à disposition :



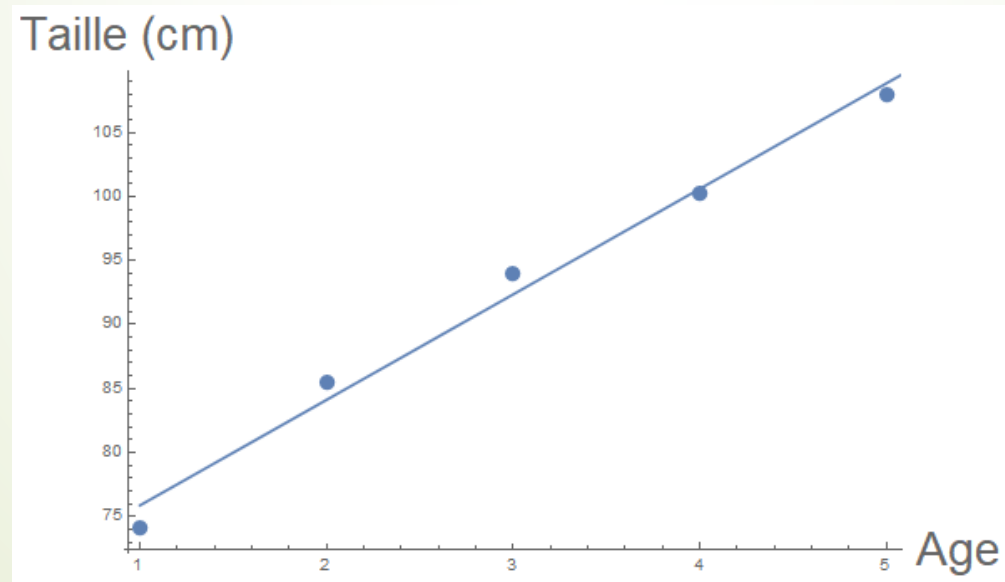
Source : <https://www.disabled-world.com/calculators-charts/height-weight-teens.php>

Machine Learning

Exemple :

Problème jouet : trouver une fonction (un programme ?) qui soit capable de prédire la taille d'un humain en fonction de son âge...

Modèle souhaité :

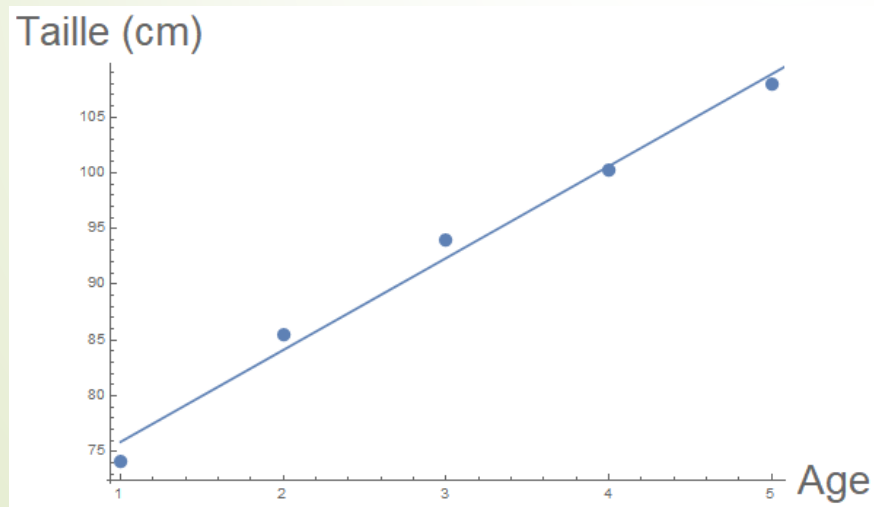


Machine Learning

Exemple :

Problème jouet : trouver une fonction (un programme ?) qui soit capable de prédire la taille d'un humain en fonction de son âge...

Modèle souhaité :



Modèle

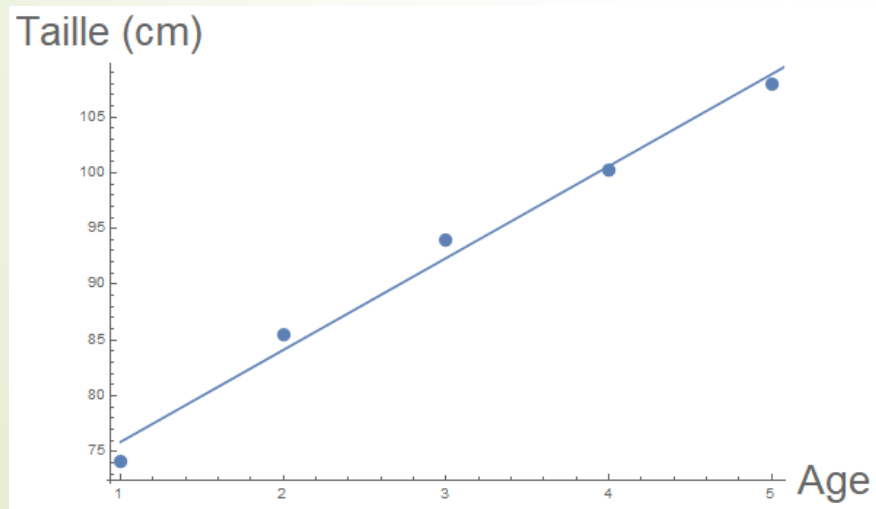
$$\text{Taille} = 8.24 \times \text{Age} + 67.64$$

Machine Learning

Exemple :

Nous avons choisi le type de fonction : $y = ax + b$

Nous avons appris (découvert à partir des données) les paramètres :
 a et b



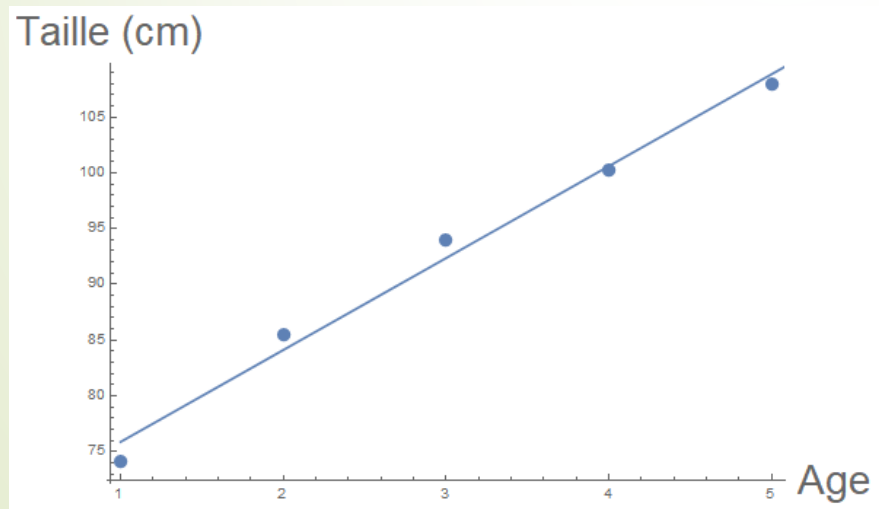
Modèle

$$\text{Taille} = 8.24 \times \text{Age} + 67.64$$

Machine Learning

Exemple :

Notre choix était-il judicieux ?



Modèle

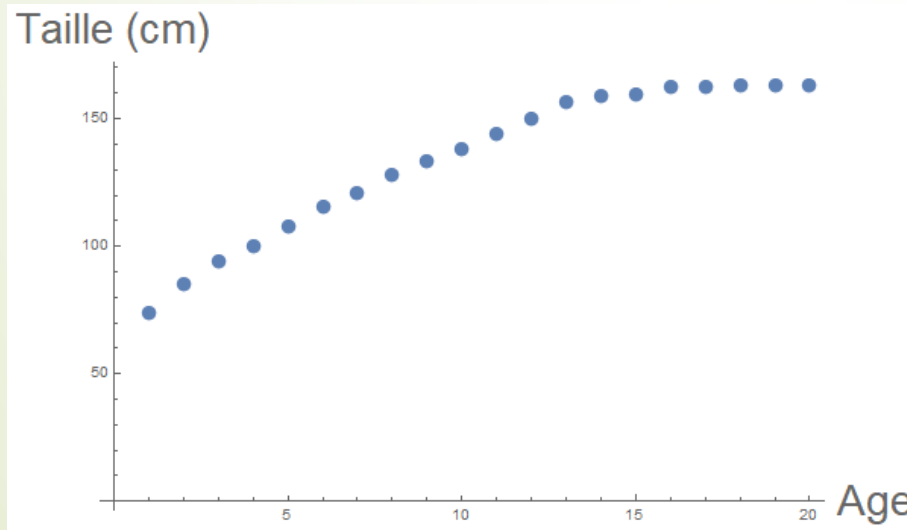
$$\text{Taille} = 8.24 \times \text{Age} + 67.64$$

Machine Learning

Exemple :

Notre choix était-il judicieux ?

Données supplémentaires à disposition :



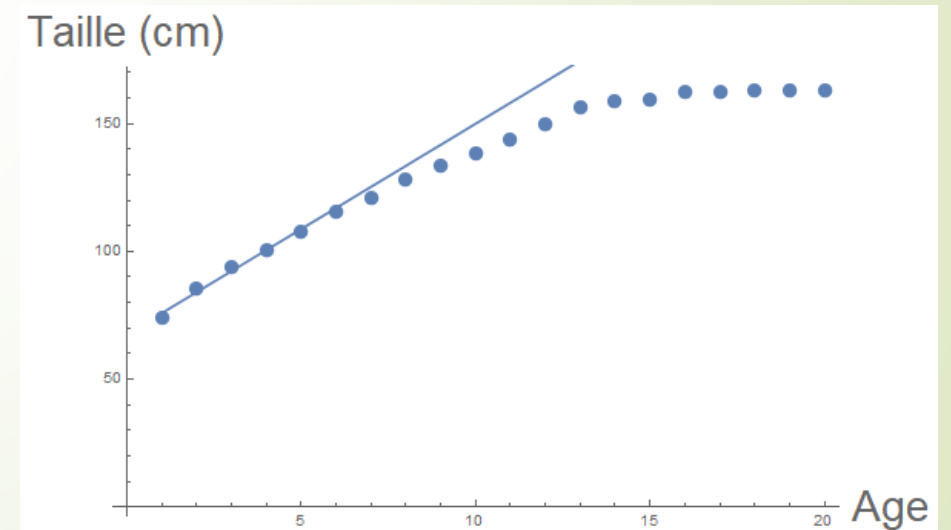
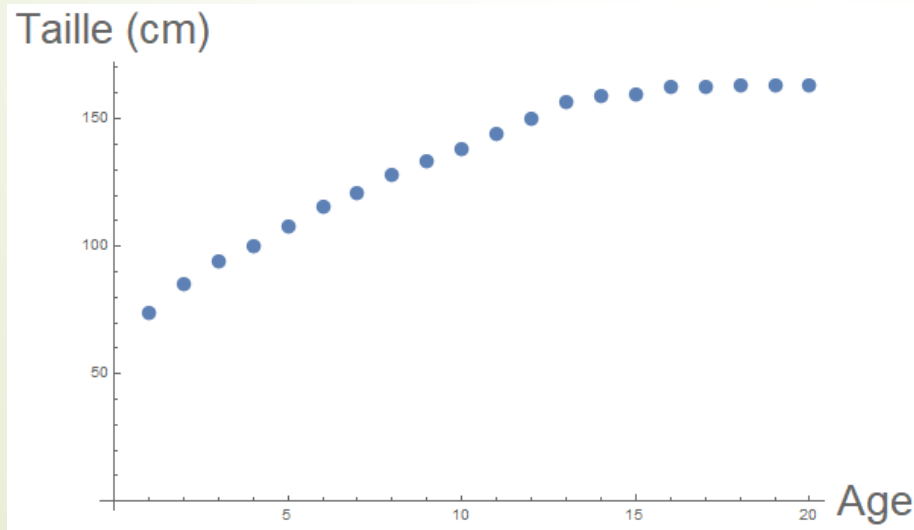
Source : <https://www.disabled-world.com/calculators-charts/height-weight-teens.php>

Machine Learning

Exemple :

Notre choix était-il judicieux ?

Données supplémentaires à disposition incompatibles avec notre précédent modèle



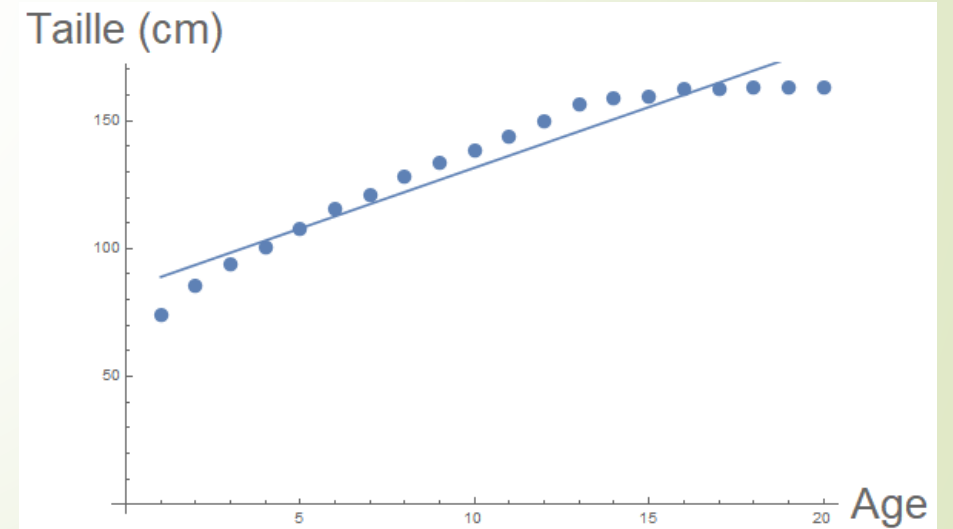
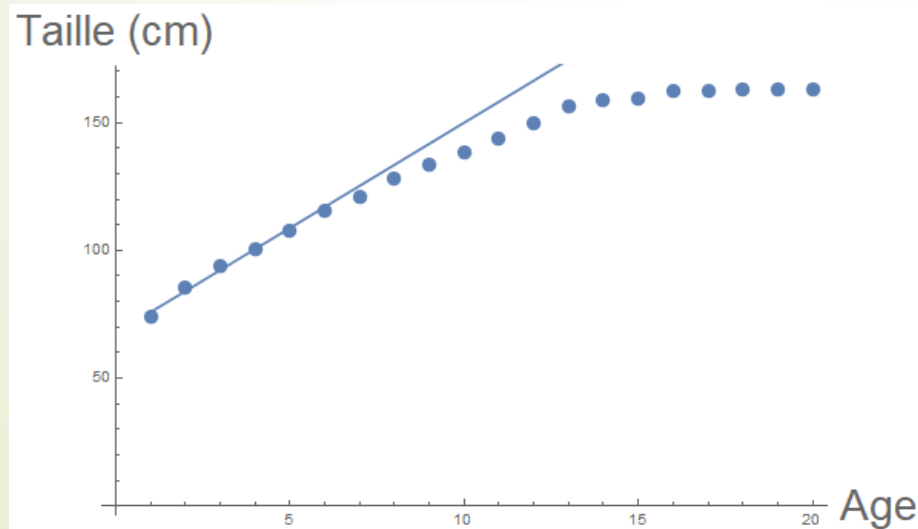
Source : <https://www.disabled-world.com/calculators-charts/height-weight-teens.php>

Machine Learning

Exemple :

Notre choix était-il judicieux ?

Mais une nouvelle régression linéaire fait à peine mieux !
Cas de Sous Apprentissage (Underfitting)



Source : <https://www.disabled-world.com/calculators-charts/height-weight-teens.php>



Machine Learning

Comment détecter l'Underfitting ?

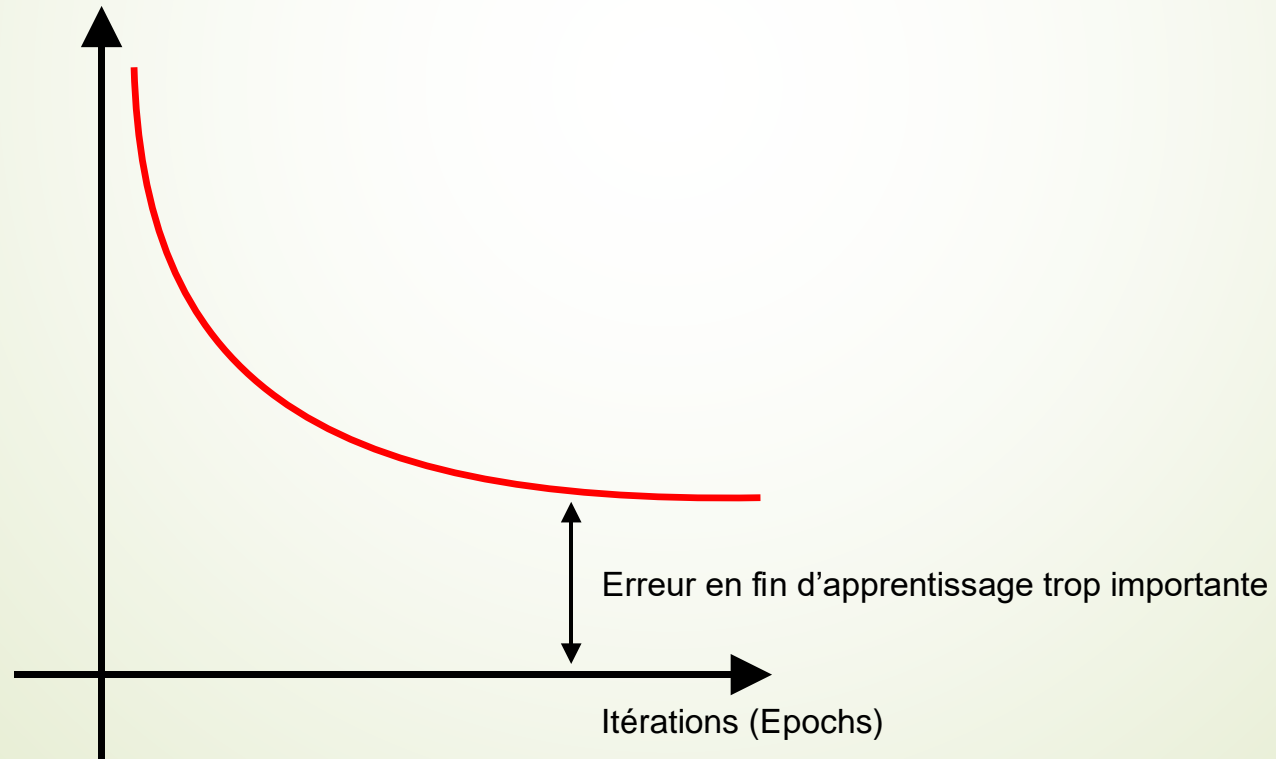


Machine Learning

Comment détecter l'Underfitting ?

Monitoring de l'erreur sur la base d'exemple au cours de l'apprentissage

Erreur sur la base d'apprentissage

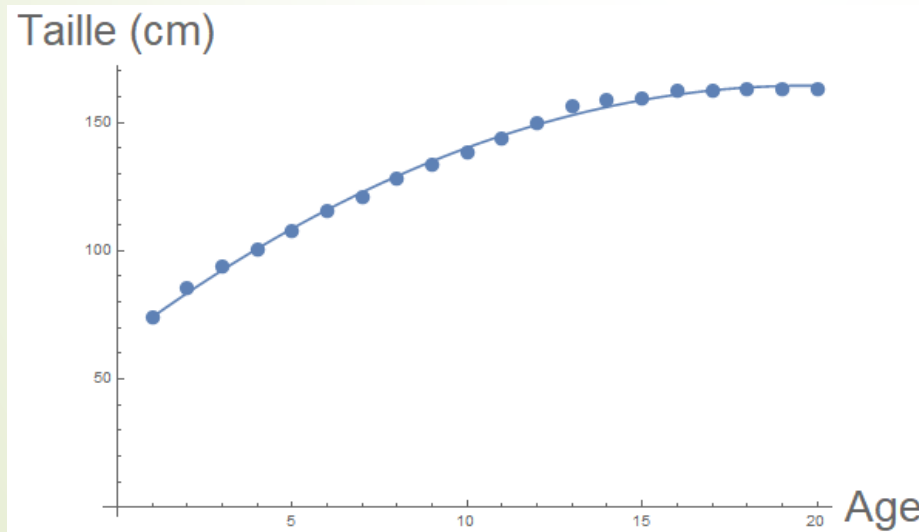


Machine Learning

Exemple :

Notre choix était-il judicieux ?

Nouveau choix de type de modèle : $y = ax^2 + bx + c$



Modèle

$$\text{Taille} = -0.259 \times \text{Age}^2 + 10.19 \times \text{Age} + 64.25$$

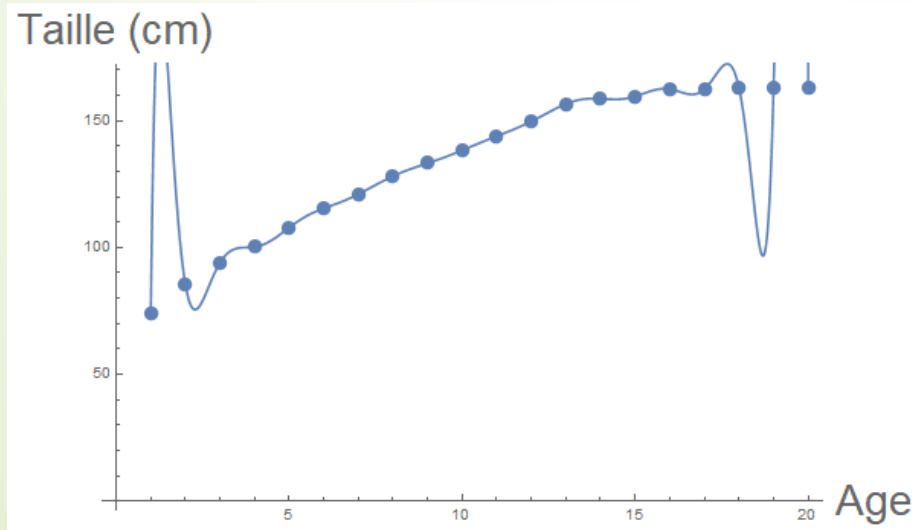
Source : <https://www.disabled-world.com/calculators-charts/height-weight-teens.php>

Machine Learning

Exemple :

Attention, un modèle puissant (grand pouvoir d'expression) n'est pas forcément meilleur !

Nouveau choix de type de modèle : polynôme de degré 22 !
Cas de Surapprentissage (Overfitting)

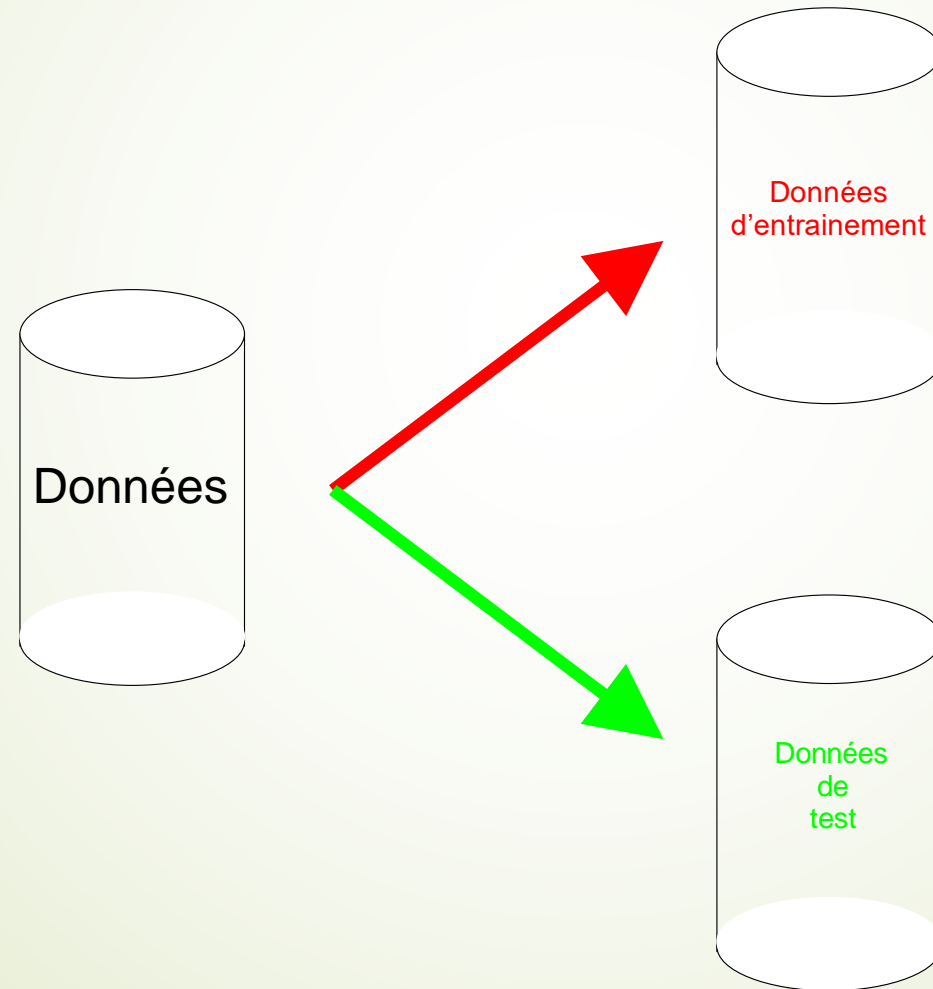


Modèle

$$\begin{aligned} \text{Taille} = & -15109.6 + 48804.6 \times \text{Age} - 66501.1 \times \text{Age}^2 + 51711.5 \times \\ & \text{Age}^3 - 25882. \times \text{Age}^4 + 8869.49 \times \text{Age}^5 - 2149.14 \times \text{Age}^6 + 372.389 \times \\ & \text{Age}^7 - 45.6506 \times \text{Age}^8 + 3.77285 \times \text{Age}^9 - 0.178969 \times \\ & \text{Age}^{10} + 0.00104904 \times \text{Age}^{11} + 0.000398215 \times \text{Age}^{12} - 0.0000153565 \times \\ & \text{Age}^{13} - 6.35198 \times 10^{-7} \times \text{Age}^{14} + 5.06753 \times 10^{-8} \times \text{Age}^{15} + 8.60159 \times 10^{-10} \\ & \times \text{Age}^{16} - 1.4655 \times 10^{-10} \times \text{Age}^{17} + 4.21198 \times 10^{-13} \times \text{Age}^{18} + 4.01457 \times 10^{-13} \\ & \times \text{Age}^{19} - 1.99247 \times 10^{-14} \times \text{Age}^{20} + 4.15134 \times 10^{-16} \times \text{Age}^{21} - \\ & 3.36896 \times 10^{-18} \times \text{Age}^{22} \end{aligned}$$

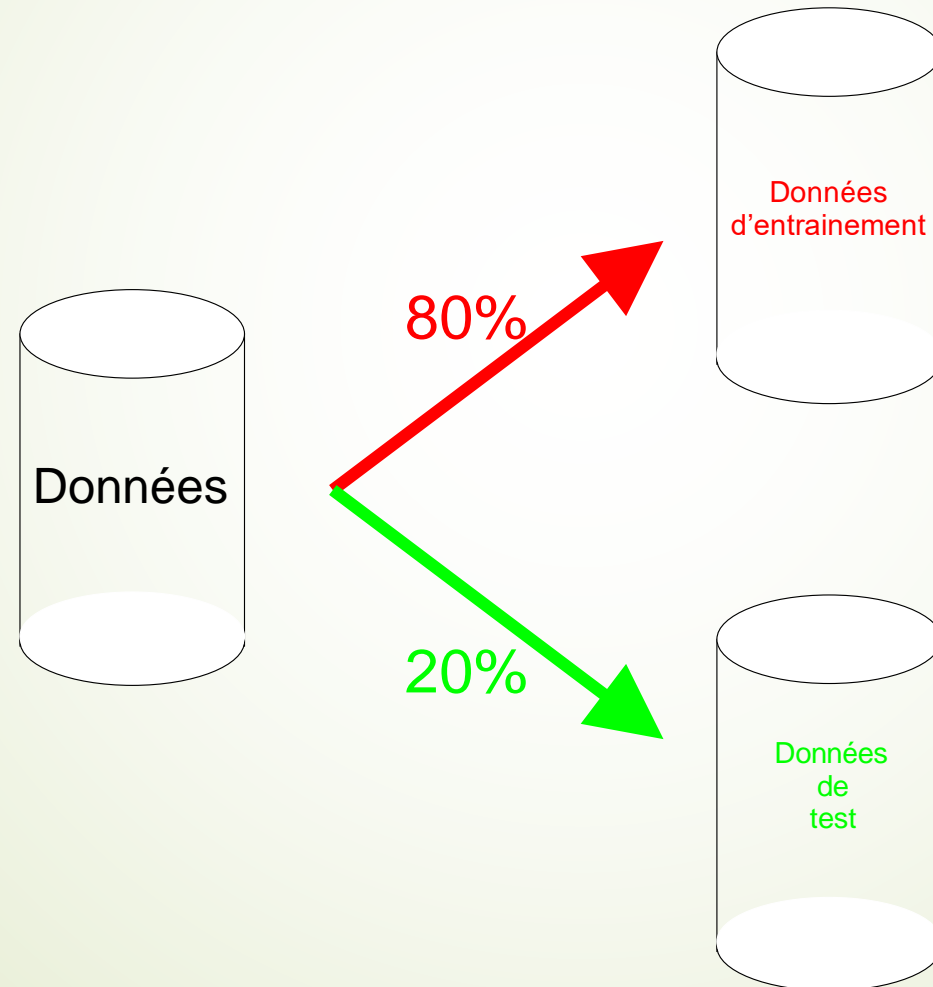
Machine Learning

Comment détecter l'Overfitting ?



Machine Learning

Comment détecter l'Overfitting ?

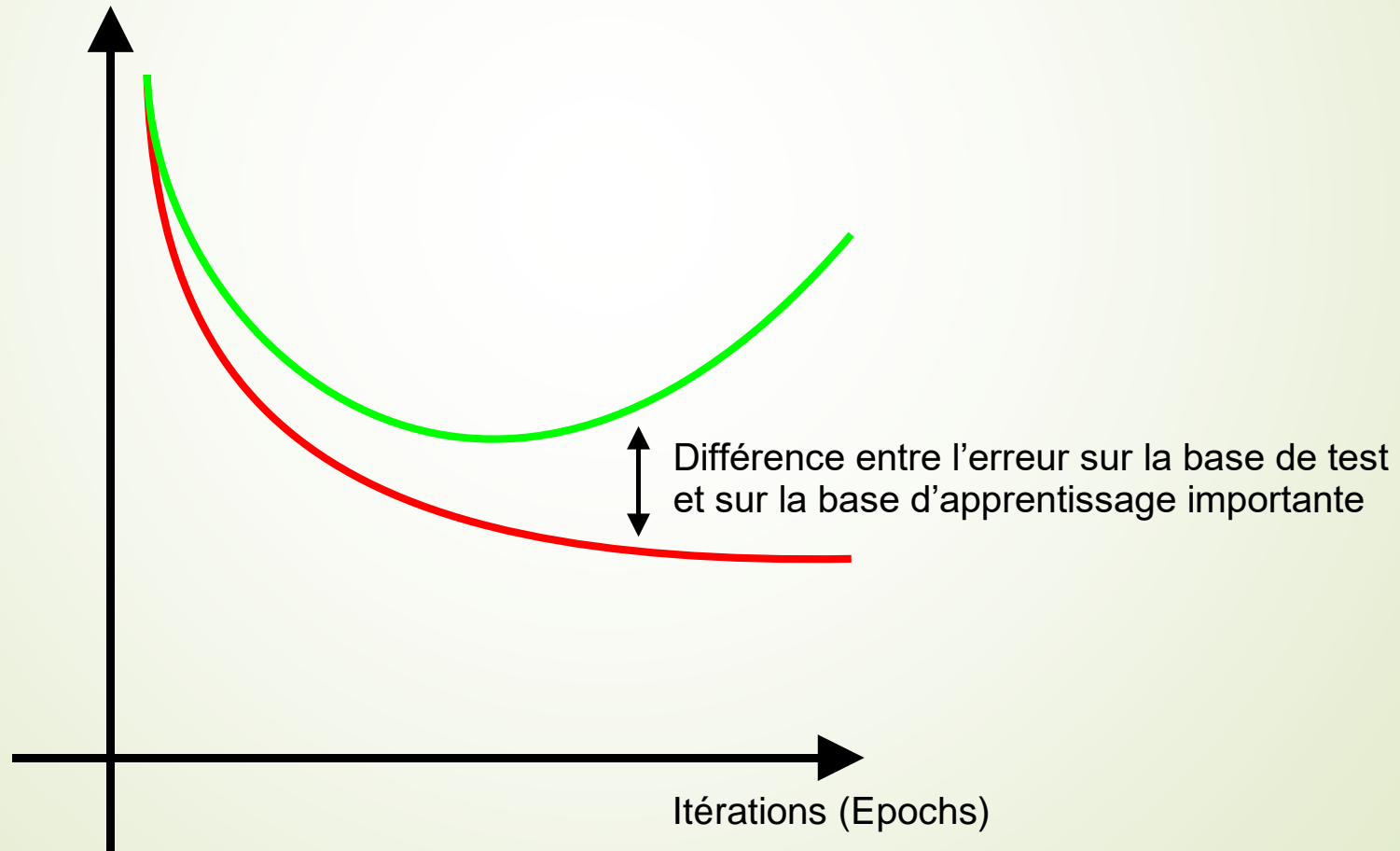


Machine Learning

Comment détecter l'Overfitting ?

Erreur sur la base de test

Erreur sur la base d'apprentissage

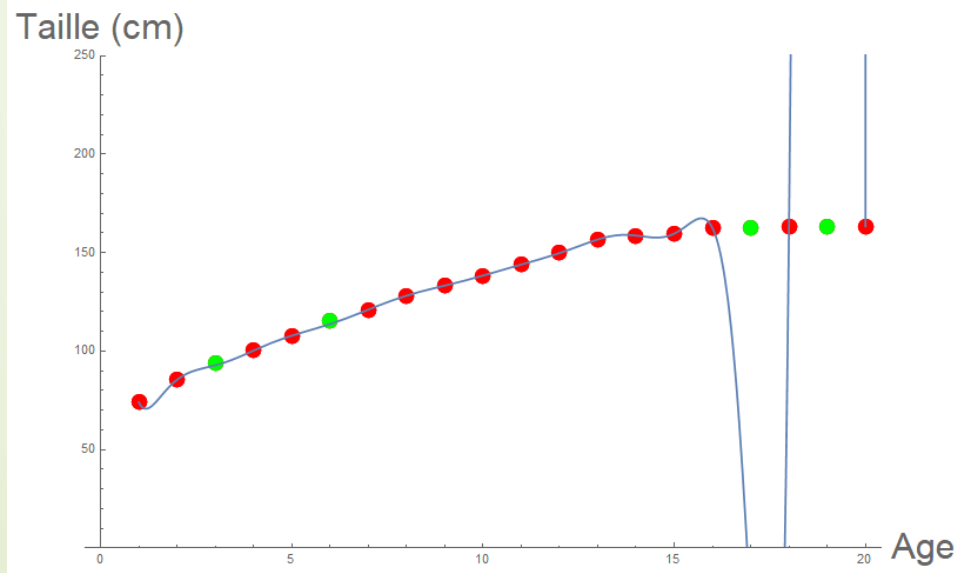


Machine Learning

Exemple :

Attention, un modèle puissant (grand pouvoir d'expression) n'est pas forcément meilleur !

Nouveau choix de type de modèle : polynôme de degré 22 !
Cas de Surapprentissage (Overfitting)



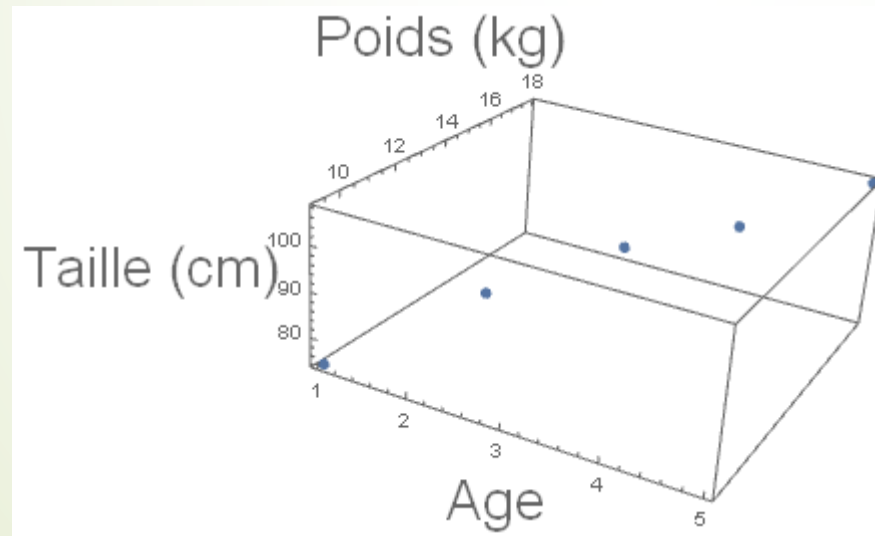
Modèle

$$\begin{aligned} \text{Taille} = & 576.416 - 1478.11 * \text{Age} + 1754.72 * \text{Age}^2 - 1122.81 * \text{Age}^3 + \\ & 436.445 * \text{Age}^4 - 108.152 * \text{Age}^5 + 17.1639 * \text{Age}^6 - 1.65478 * \\ & \text{Age}^7 + 0.0772617 * \text{Age}^8 + 0.000774736 * \text{Age}^9 - 0.000217763 * \\ & \text{Age}^{10} - 1.44252 * 10^{-7} * \text{Age}^{11} + \\ & 5.57711 * 10^{-7} * \text{Age}^{12} + 7.70546 * 10^{-9} * \text{Age}^{13} - 1.28263 * 10^{-9} * \\ & \text{Age}^{14} - 5.29395 * 10^{-11} * \text{Age}^{15} + 1.82361 * 10^{-12} * \text{Age}^{16} + \\ & 1.98712 * 10^{-13} * \text{Age}^{17} + 1.22125 * 10^{-15} * \text{Age}^{18} - 5.31121 * 10^{-16} * \\ & \text{Age}^{19} - 1.50563 * 10^{-17} * \text{Age}^{20} + 1.75709 * 10^{-18} * \text{Age}^{21} - \\ & 3.175 * 10^{-20} * \text{Age}^{22} \end{aligned}$$

Machine Learning

Doit-on se limiter à une seule entrée ?

Ex : Taille en fonction de l'âge et du poids



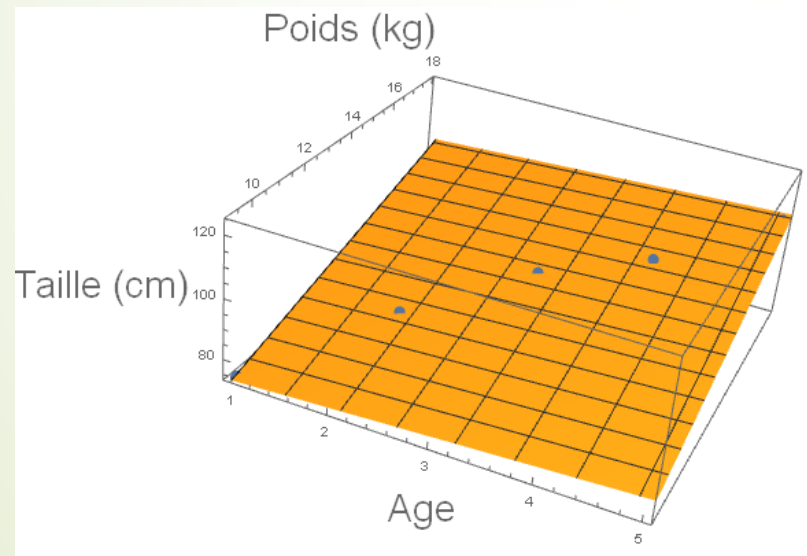
Source : <https://www.disabled-world.com/calculators-charts/height-weight-teens.php>

Machine Learning

Doit-on se limiter à une seule entrée ?

Ex : Taille en fonction de l'âge et du poids

Choix de type de modèle : $y = ax + bz + c$



Modèle

$$\text{Taille} = 1.74 \times \text{Age} + 3.13 \times \text{Poids} + 44.19$$



Machine Learning

Doit-on se limiter à une seule entrée ?

Ex : Taille en fonction de l'âge et du poids

Et pourquoi pas : $y = ax + bz + cx^2 + dz^2 + exz + f$?



Machine Learning

Doit-on se limiter à une seule entrée ?

Ex : Taille en fonction de l'âge et du poids

Et pourquoi pas : $y = ax + bz + cx^2 + dz^2 + exz + f$?

Ou encore : $y = ax + be^z + c\sqrt{x} + d \sin(z) + e \cos(xz) + f$?

Machine Learning

Doit-on se limiter à une seule entrée ?

Ex : Taille en fonction de l'âge et du poids

Et pourquoi pas : $y = ax + bz + cx^2 + dz^2 + exz + f$?

Ou encore : $y = ax + be^z + c\sqrt{x} + d \sin(z) + e \cos(xz) + f$?

Ou encore : $y = \text{if } x > b \text{ then } c \text{ elif } z \leq d \text{ then } e \text{ else } f$?



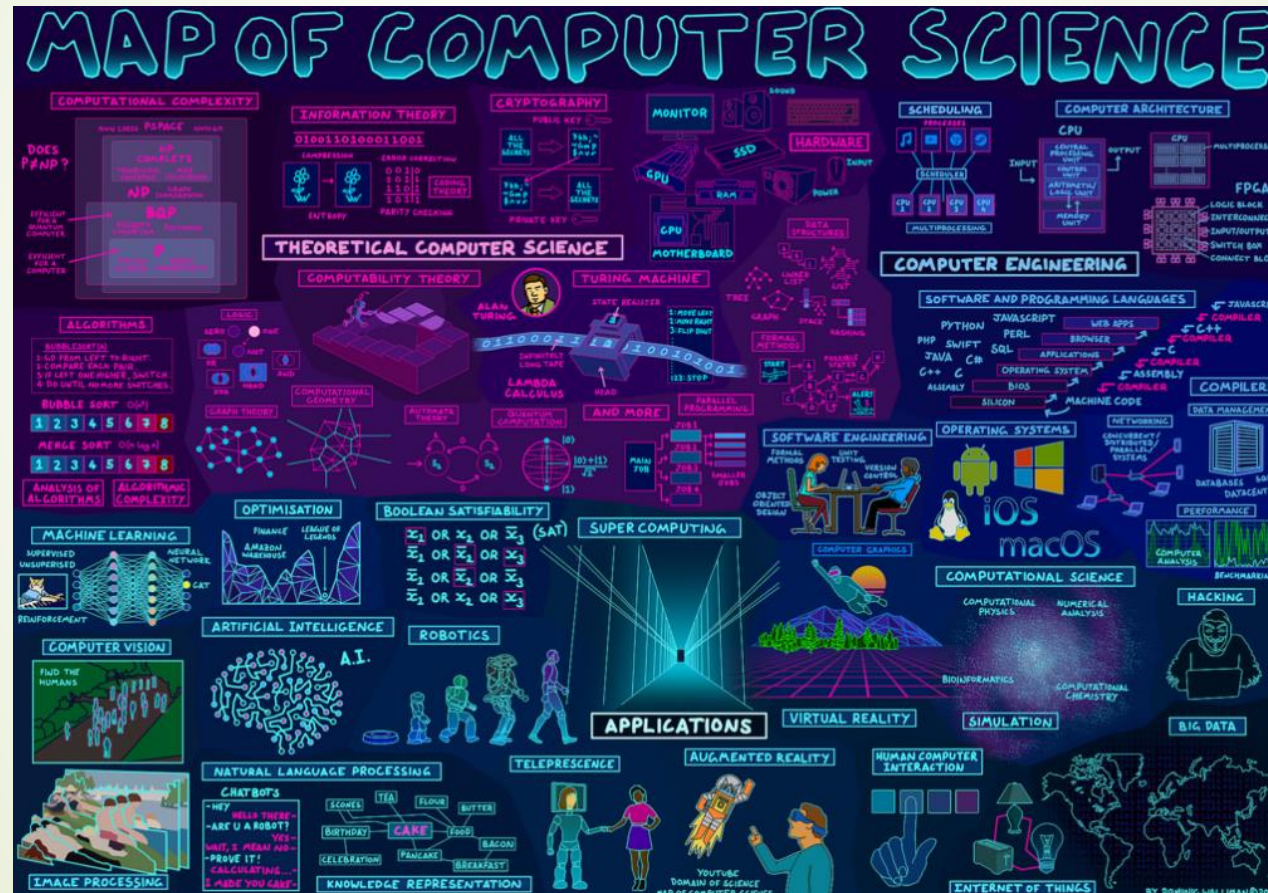
Petite parenthèse ...

Qui ici est déjà en overdose de notations mathématiques ?



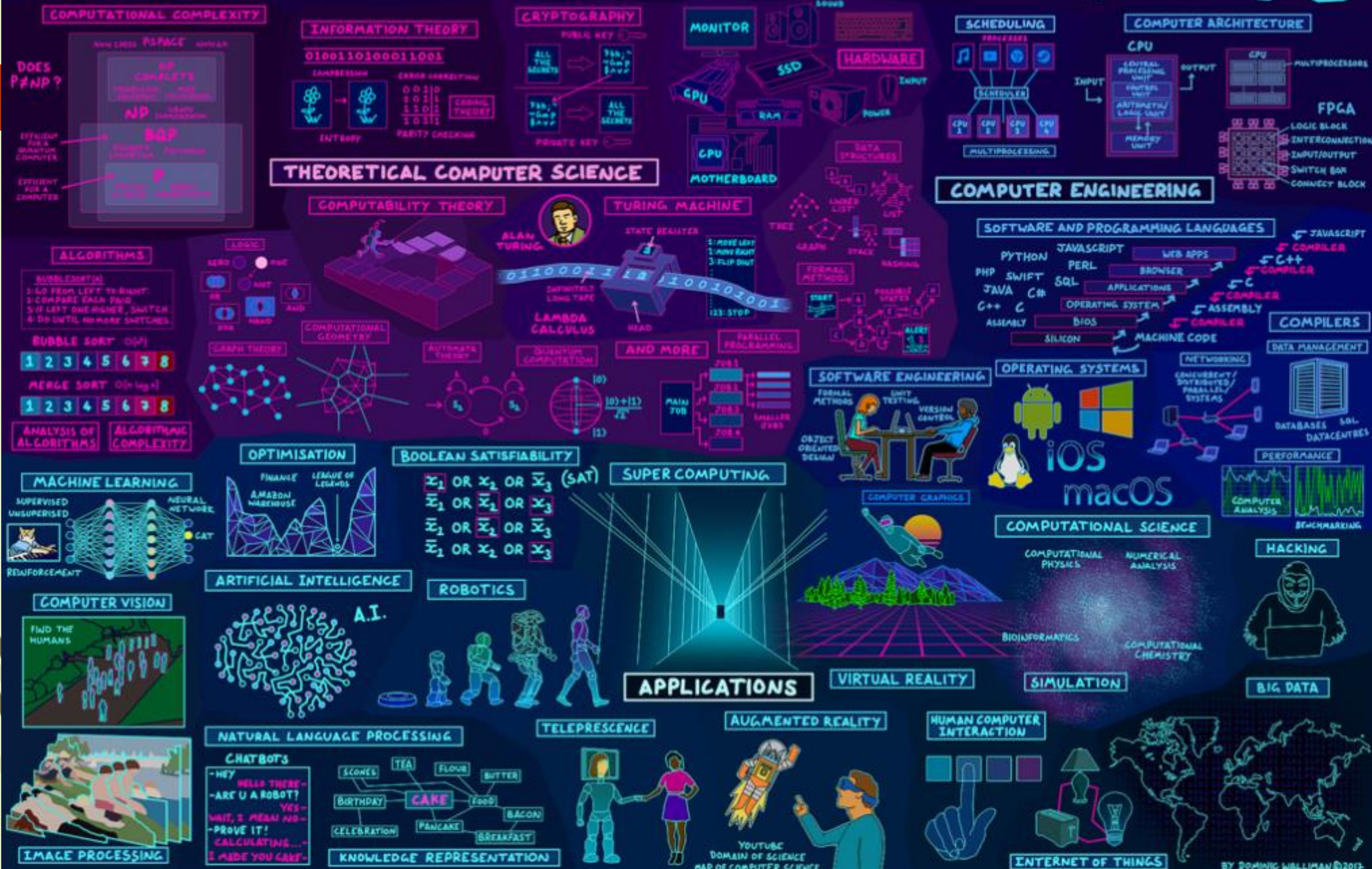
Petite parenthèse ...

Quelques illustrations de nos disciplines ... (Dominic Walliman)



Source : <https://www.flickr.com/photos/95869671@N08/>

THE THEORETICAL COMPUTER SCIENCE



N'oubliez pas la vue d'ensemble ;)



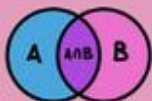
FOUNDATIONS

FUNDAMENTAL RULES

MATHEMATICAL LOGIC

$$p \Rightarrow q$$

SET THEORY



CONSISTENT SET OF AXIOMS?
GÖDEL INCOMPLETENESS THEOREMS
NOPE



CATEGORY THEORY

THEORY OF COMPUTATION

00011100

P ≠ NP?

COMPLEXITY THEORY

1 1 1
1 2 1
1 3 3 1
1 4 6 4 1

NUMBER THEORY

PARTITION THEORY

GROUP THEORY

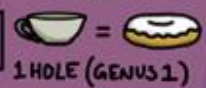


PERMUTATION GROUP

MEASURE THEORY



TOPOLOGY



DIFFERENTIAL GEOMETRY

COMPLEX ANALYSIS

FRactal Geometry

MANDELBROT SET



DYNAMICAL SYSTEMS

FLUID FLOW



ECOSYSTEMS

CHAOS THEORY



CARDINAL NUMBERS

\aleph_0 ALEPH NULL

OCTONION

$\{e_0, e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$

QUATERNION

$a + bi + cj + dk$

PI π

EXPONENTIAL e

COMPLEX NUMBERS

$3, i, 4 + 3i, -4i$

REAL NUMBERS

$-4\pi, \sqrt{2}, e$

RATIONAL NUMBERS

$-7, \frac{1}{2}, 2.32$

NATURAL NUMBERS

$1, 2, 3, 4, 5, \dots$

INTEGERS

$\dots -2, -1, 0, 1, 2, \dots$

ARITHMETIC $+$ $-$ \times \div

COMBINATORICS



GRAPH THEORY



LINEAR ALGEBRA

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 17 & 26 \\ 38 & 47 \end{bmatrix}$$

MATRICES

$$\begin{pmatrix} 6 & 7 \\ -3 & 2i \end{pmatrix}$$

ALGEBRA

$$\begin{aligned} x^2 - 4x - 8 &= 5x + 12 \\ x^2 - 9x - 36 &= 0 \\ (x+3)(x-12) &= 0 \end{aligned}$$

EQUATION

$$y = mx + c$$

VECTORS

$$\vec{x}$$

STRUCTURES

SPACES

GEOMETRY



CHANGES

CALCULUS

$$\text{GRADIENT} = \frac{dy}{dx}$$

INTEGRAL

$$\text{AREA} = \int_1^9 f(x) dx$$

DIFFERENTIAL EQUATIONS

VECTOR CALCULUS



CRYPTOGRAPHY

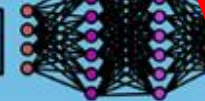


COMPUTER SCIENCE

TURING MACHINE

```
while awake:
  do_science()
  if self.tired():
    awake = False
    self.repair_brain()
```

MACHINE LEARNING



OPTIMIZATION

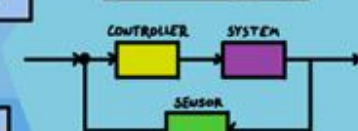
STATISTICS

MATHEMATICAL ECONOMICS

ECONOMICS

APPLIED MATHEMATICS

ENGINEERING

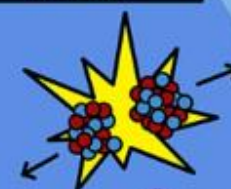


CONTROL THEORY

BIOMATHEMATICS

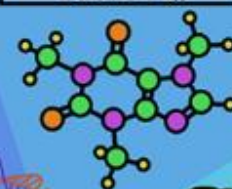


MATHEMATICAL PHYSICS



THEORETICAL PHYSICS

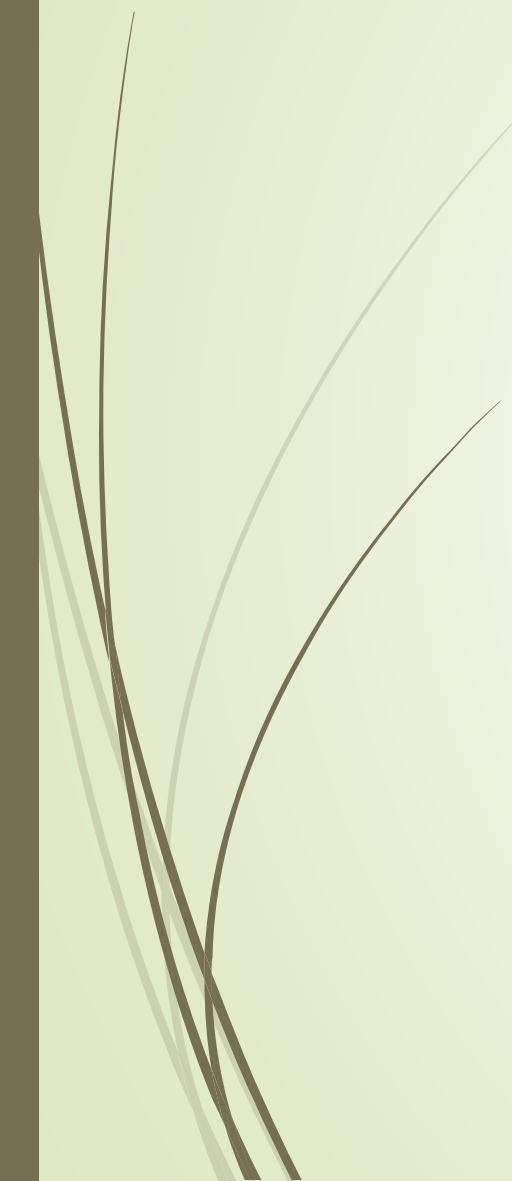
MATHEMATICAL CHEMISTRY





Petite parenthèse ...

Mes conseils pour digérer quelques formules obscures :

- Ne partez pas en courant
 - Ayez la certitude qu'en y passant suffisamment de temps, vous aurez l'épiphanie au bout du compte !
 - Laissez votre cerveau travailler la nuit !
- 

Parenthèse imbriquée ...

Pour travailler votre résilience aux casse-têtes et vous habituer aux épiphanies :



Machine Learning ... Reprenons ...

Doit-on se limiter à une seule entrée ?

Ex : Taille en fonction de l'âge et du poids

Et pourquoi pas : $y = ax + bz + cx^2 + dz^2 + exz + f$?

Ou encore : $y = ax + be^z + c\sqrt{x} + d \sin(z) + e \cos(xz) + f$?

Ou encore : $y = \text{if } x > b \text{ then } c \text{ elif } z \leq d \text{ then } e \text{ else } f$?



Deep Learning



Deep Learning

Théorème d'approximation universelle

Cybenko 89

Pour toute fonction F continue définie et bornée sur un ensemble borné, et pour tout ε , il existe un réseau de neurones à 1 couche cachée de neurones sigmoïdes qui approxime F à ε près.

Sussman 92

Les réseaux à une couche cachée forment une famille d'approximateurs parcimonieux : à nombre égal de paramètres on approxime correctement plus de fonctions qu'avec des polynômes.



Deep Learning

Théorème d'approximation universelle

Cybenko 89

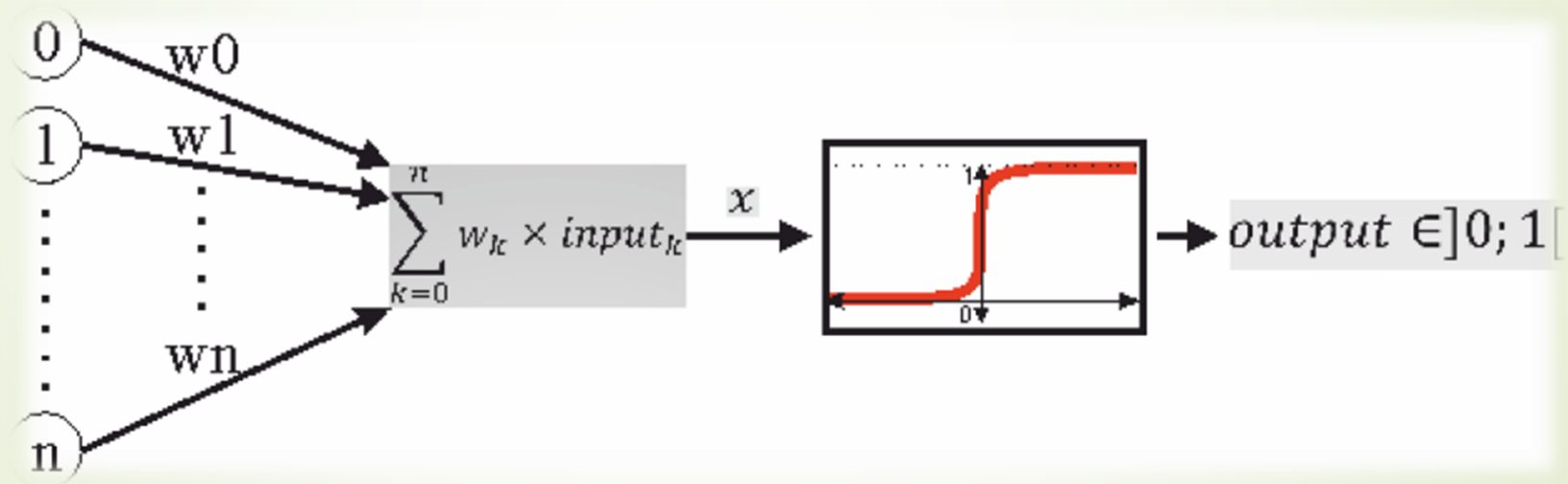
Pour toute fonction F continue définie et bornée sur un ensemble borné, et pour tout ε , il existe un réseau de neurones à 1 couche cachée de **neurones sigmoïdes** qui approxime F à ε près.

Sussman 92

Les réseaux à une couche cachée forment une famille d'approximateurs parcimonieux : à nombre égal de paramètres on approxime correctement plus de fonctions qu'avec des polynômes.

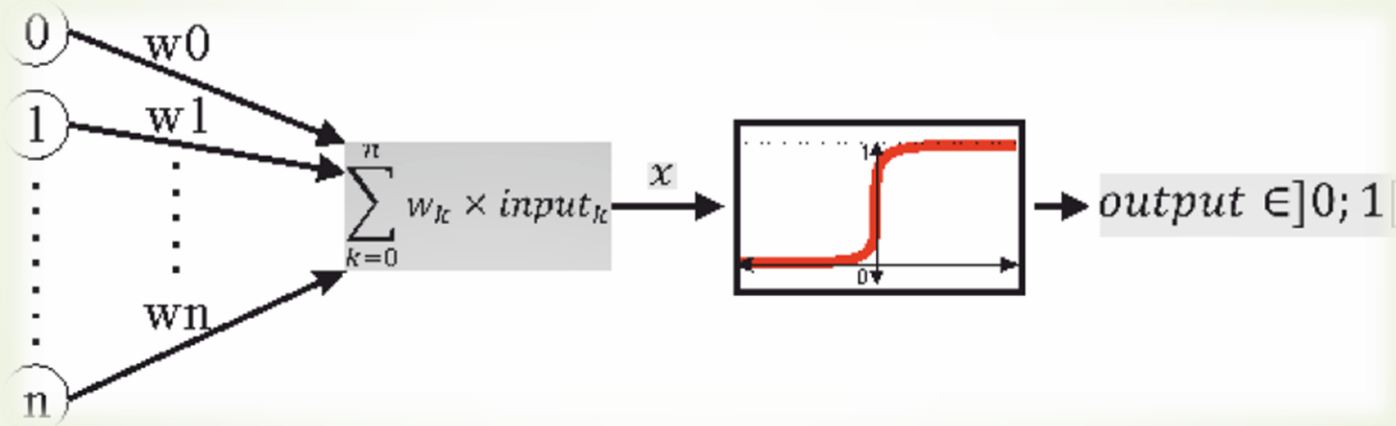
Deep Learning

Neurone artificiel avec fonction d'activation sigmoïde



Deep Learning

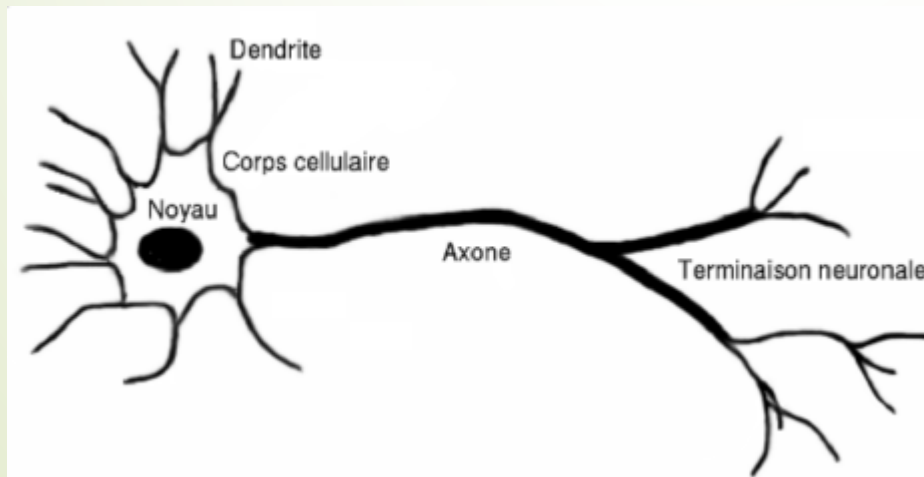
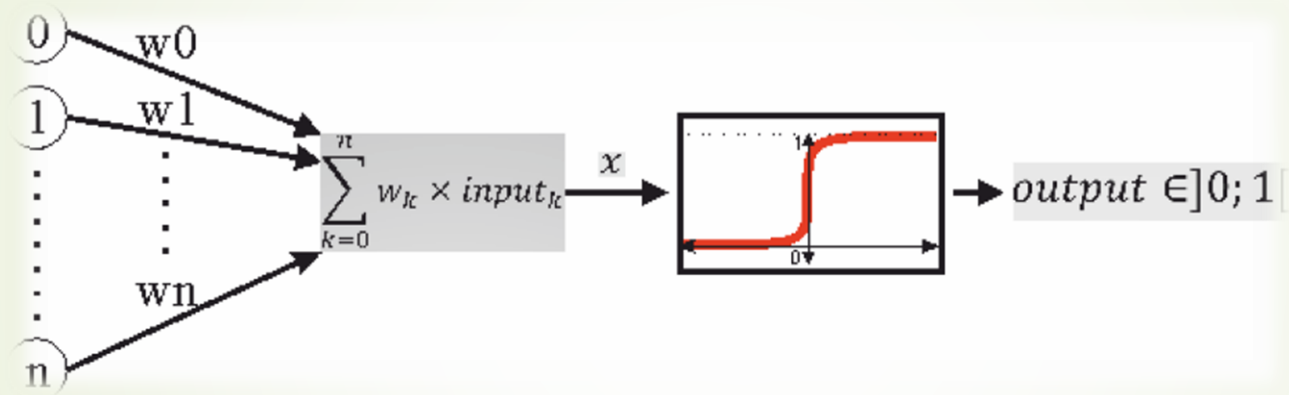
Neurone artificiel avec fonction d'activation sigmoïde



$$output = \frac{1}{1 + e^{-\sum_{k=0}^n w_k x_k}}$$

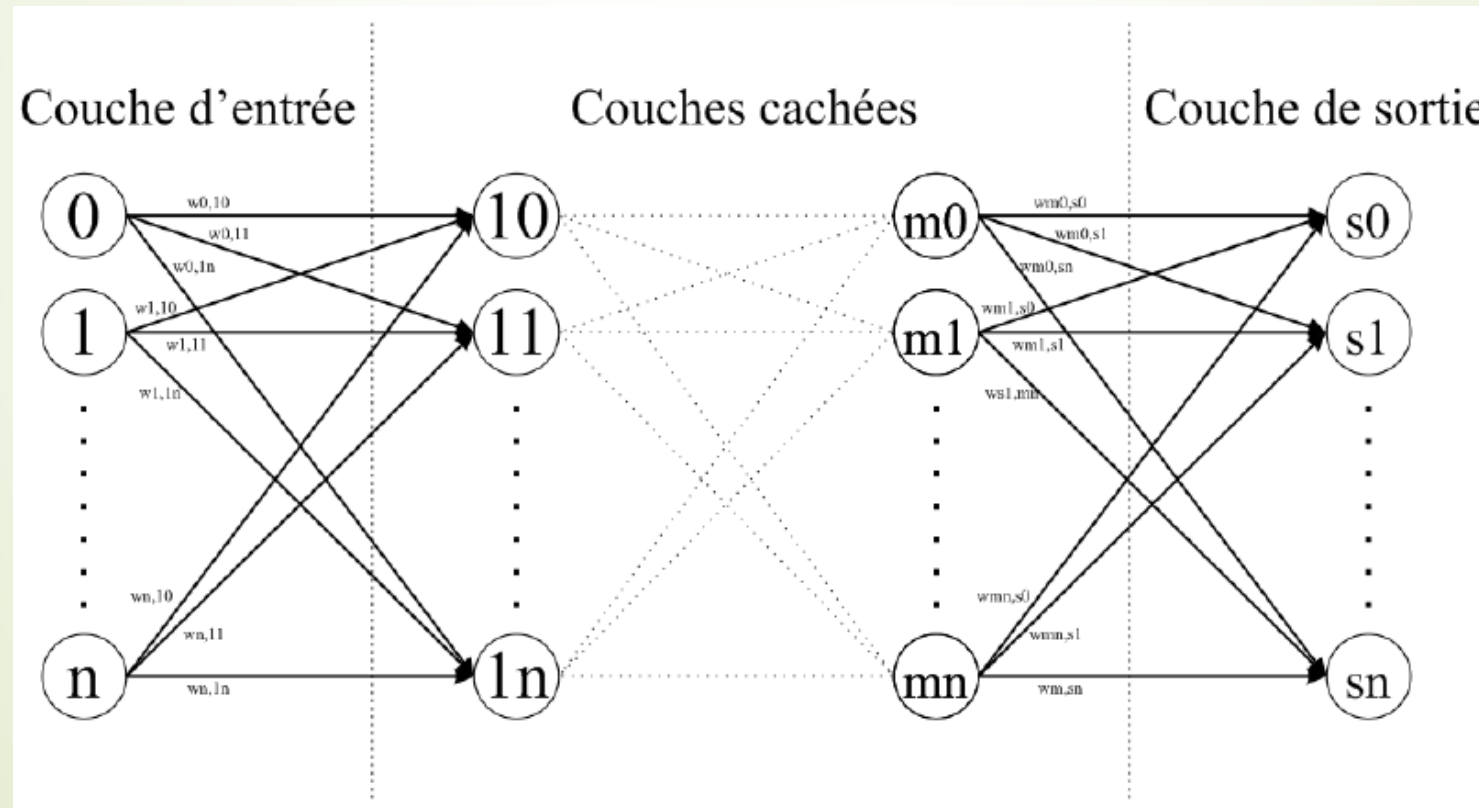
Deep Learning

Métaphore Biologique



Deep Learning

Réseau de Neurones artificiels avec fonction d'activation sigmoïde (ici un PMC)



Deep Learning

Démystification ...

Basic Math

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}$$

Dangerous Artificial Intelligence

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix} * \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix} * \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix} * \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}$$

Deep Learning

Démystification ...

Basic Math

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}$$

Dangerous Artificial Intelligence

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix} * \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix} * \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix} * \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}$$

Deep Learning

Exemple de problème de classification :
le 'Ou' exclusif (XOR)
(la nécessité d'utiliser des couches cachées ...)

X	Y	XOR(X, Y)
True	True	False
True	False	True
False	True	True
False	False	False

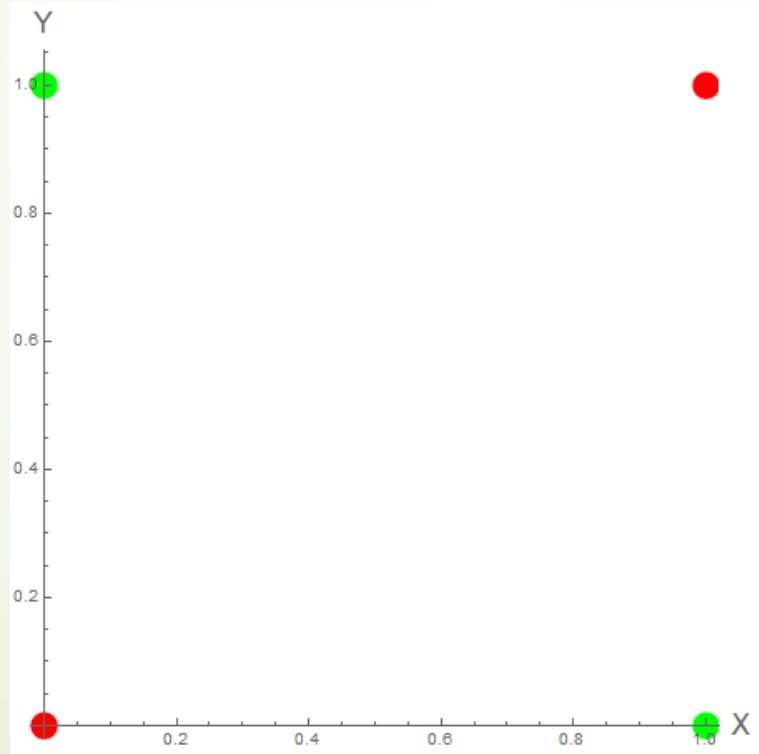
Deep Learning

Exemple de problème de classification :
le 'Ou' exclusif (XOR)
(la nécessité d'utiliser des couches cachées ...)

X	Y	XOR(X, Y)
1	1	0
1	0	1
0	1	1
0	0	0

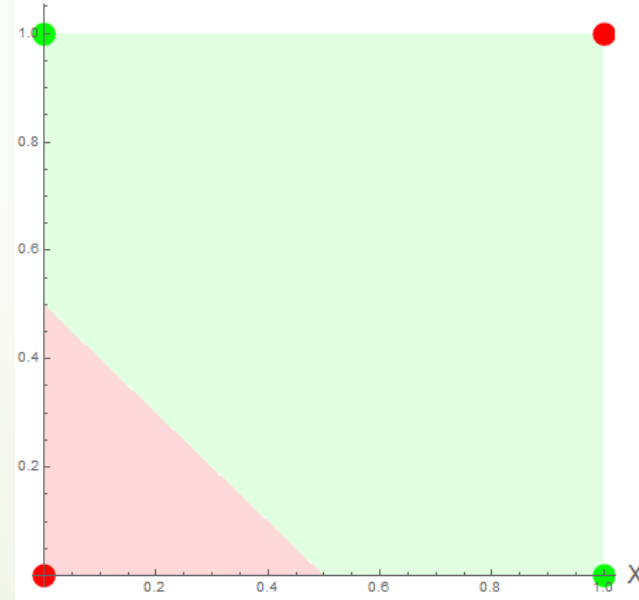
Deep Learning

Exemple de problème de classification :
le 'Ou' exclusif (XOR)
(la nécessité d'utiliser des couches cachées ...)



Deep Learning

Exemple de problème de classification :
le 'Ou' exclusif (XOR)
(la nécessité d'utiliser des couches cachées ...)

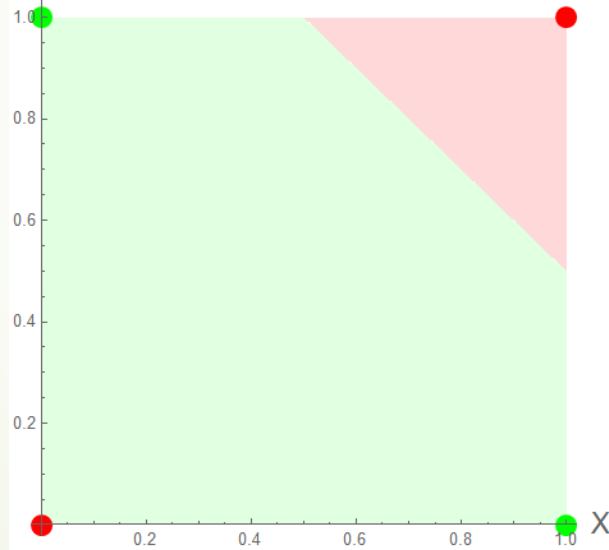


$Output1 = \text{if } 1 \times x + 1 \times y + (-0,5) \geq 0 \text{ then } 1 \text{ else } 0$

$Output1 \approx \text{sigmoid}(1 \times x + 1 \times y + (-0,5))$

Deep Learning

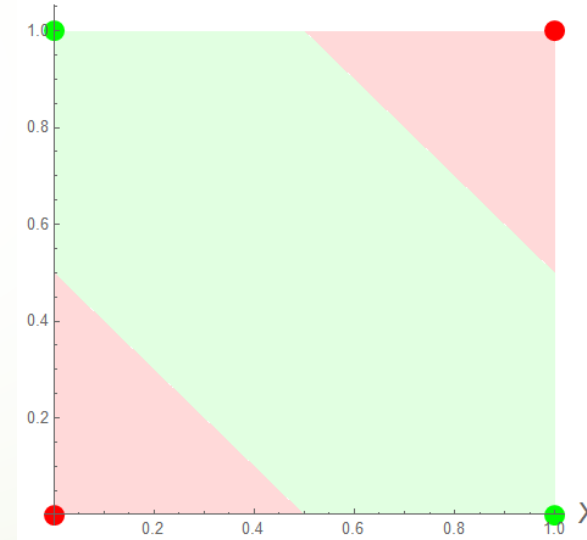
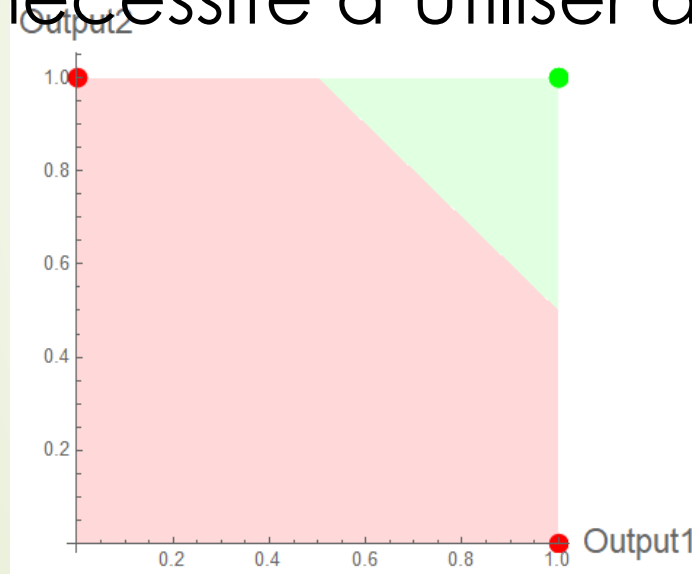
Exemple de problème de classification :
le 'Ou' exclusif (XOR)
(la nécessité d'utiliser des couches cachées ...)



$$\text{Output2} = \text{if } (-1) \times x + (-1) \times y + 1,5 \geq 0 \text{ then } 1 \text{ else } 0$$
$$\text{Output2} \approx \text{sigmoid}((-1) \times x + (-1) \times y + 1,5)$$

Deep Learning

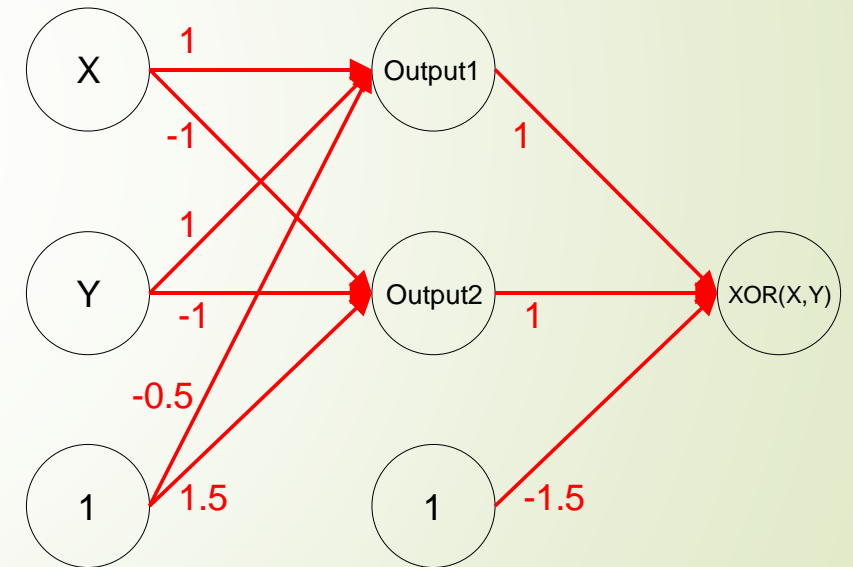
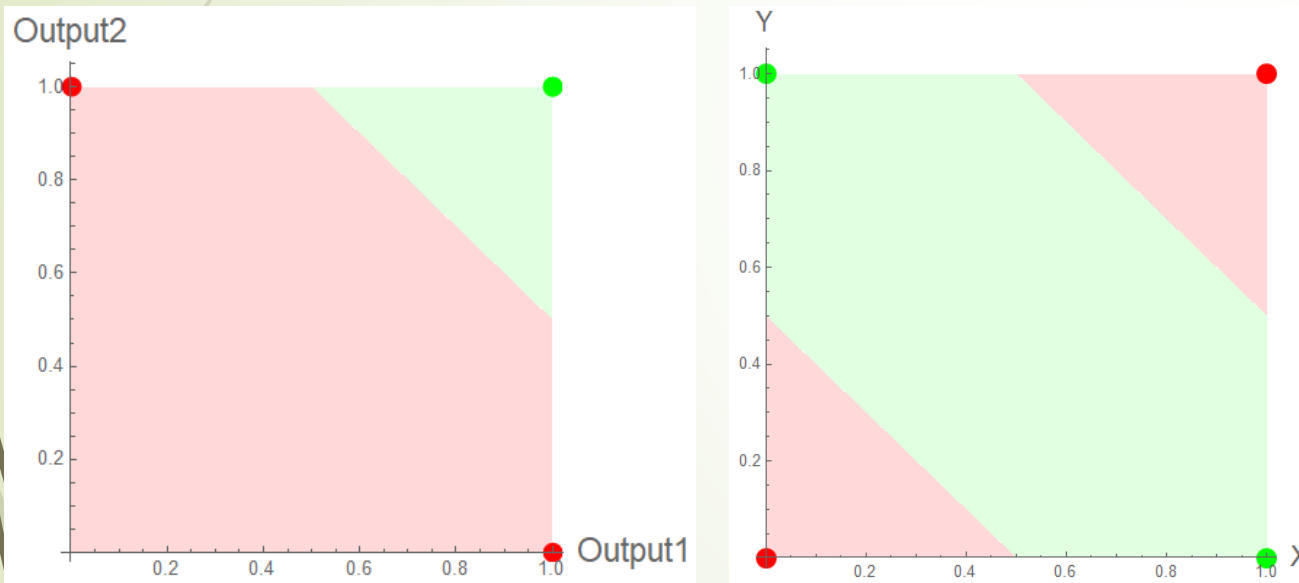
Exemple de problème de classification :
le 'Ou' exclusif (XOR)
(la nécessité d'utiliser des couches cachées ...)



$$XOR(x, y) = \text{if } 1 \times \text{Output1} + 1 \times \text{Output2} + (-1,5) \geq 0 \text{ then } 1 \text{ else } 0$$
$$XOR(x, y) \approx \text{sigmoid}(1 \times \text{Output1} + 1 \times \text{Output2} + (-1,5))$$

Deep Learning

Exemple de problème de classification :
le 'Ou' exclusif (XOR)
(la nécessité d'utiliser des couches cachées ...)



$$XOR(x, y) = \text{if } 1 \times \text{Output1} + 1 \times \text{Output2} + (-1,5) \geq 0 \text{ then } 1 \text{ else } 0$$
$$XOR(x, y) \approx \text{sigmoid}(1 \times \text{Output1} + 1 \times \text{Output2} + (-1,5))$$



Deep Learning

Apprendre une projection des données dans un espace où le problème devient plus simple à résoudre (données linéairement séparables ou régression linéaire possible par exemple) **ET** apprendre à résoudre le problème dans cet espace.



Deep Learning

Machine Learning Classique :

- Un expert choisit les combinaisons de colonnes pertinentes de la base de données (projection)
- Apprentissage d'un modèle simple à partir de ces choix

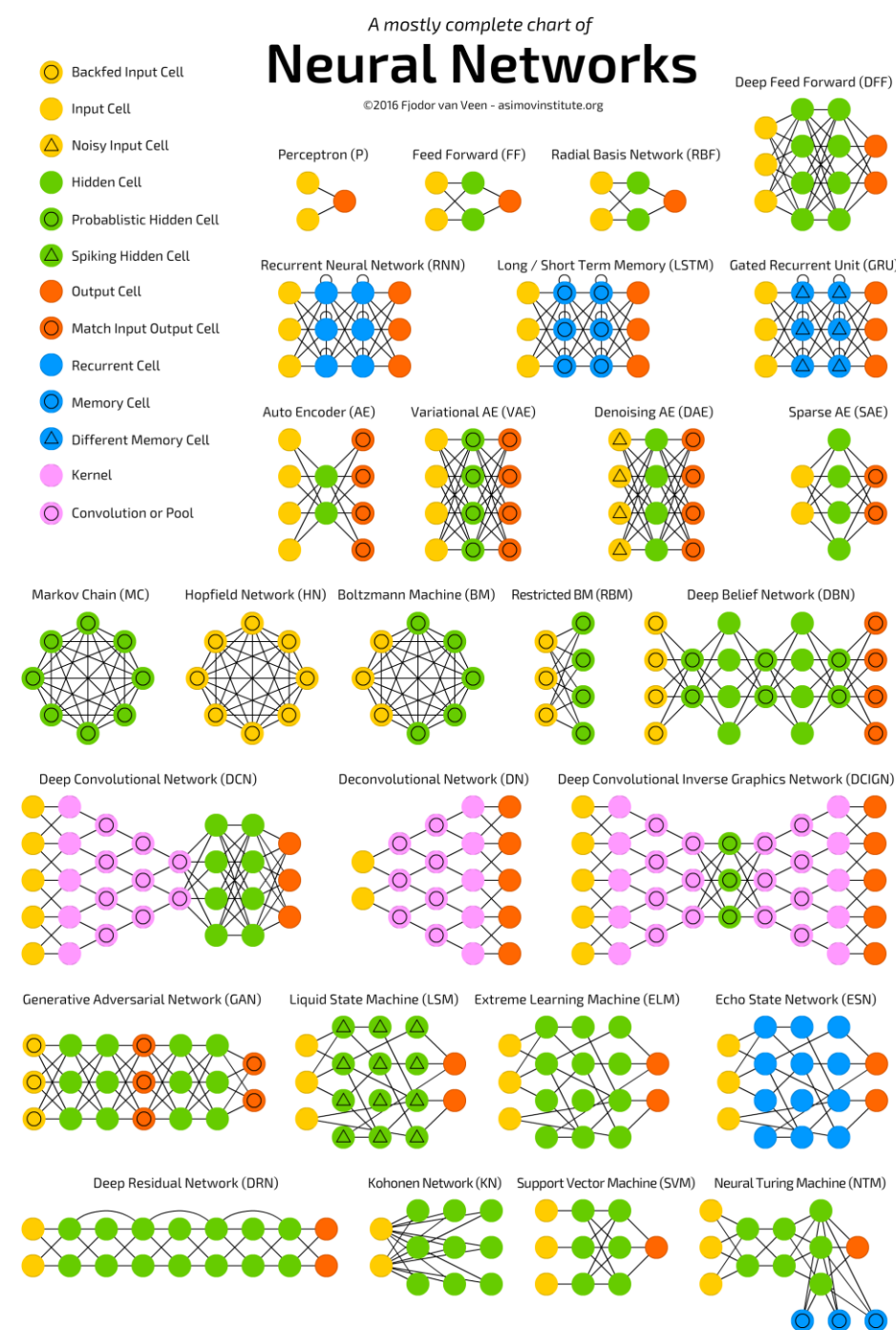
Deep Learning :

- Le modèle apprend 'tout' depuis les données 'brutes'

Deep Learning

Il n'y a pas que le
Perceptron Multi
Couches (PMC)
dans les réseaux
de neurones
artificiels !

Source : <http://www.asimovinstitute.org/neural-network-zoo/>





Machine Learning

Distinctions ...

Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension
- ...

Machine Learning

Distinctions ...

Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension
- ...



Machine Learning

Distinctions ...

Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension
- ...



Machine Learning

Distinctions ...

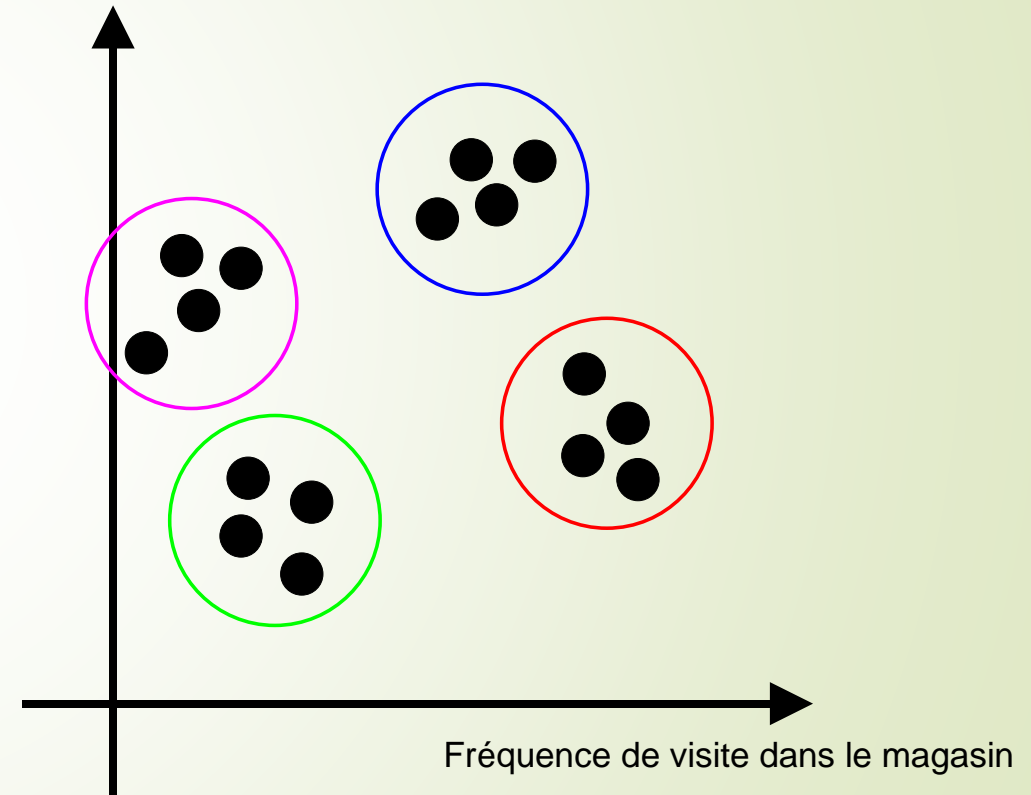
Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension
- ...

Somme d'argent dépensée dans le magasin



Machine Learning

Distinctions ...

Supervisé

- Classification
- Régression



Non Supervisé

- Clustering
- Génération
- Réduction de dimension
- ...



Machine Learning

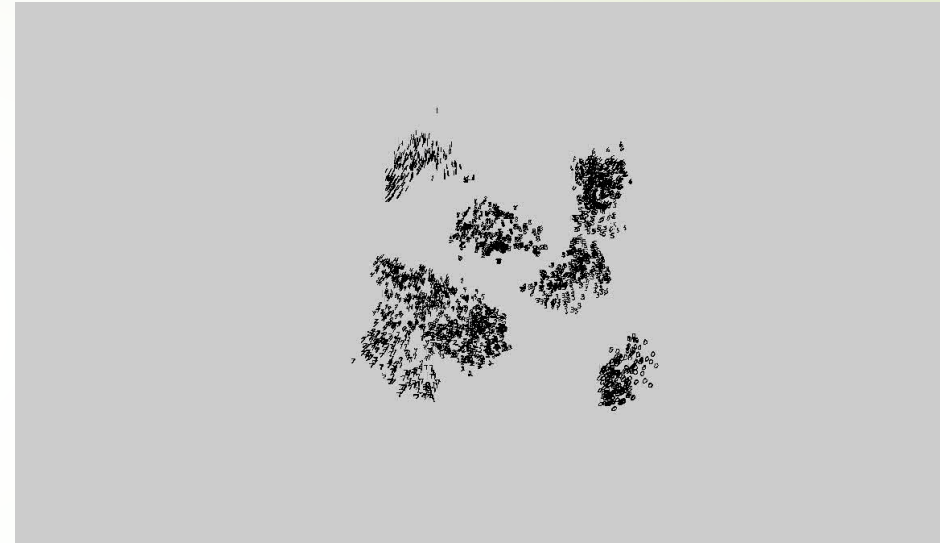
Distinctions ...

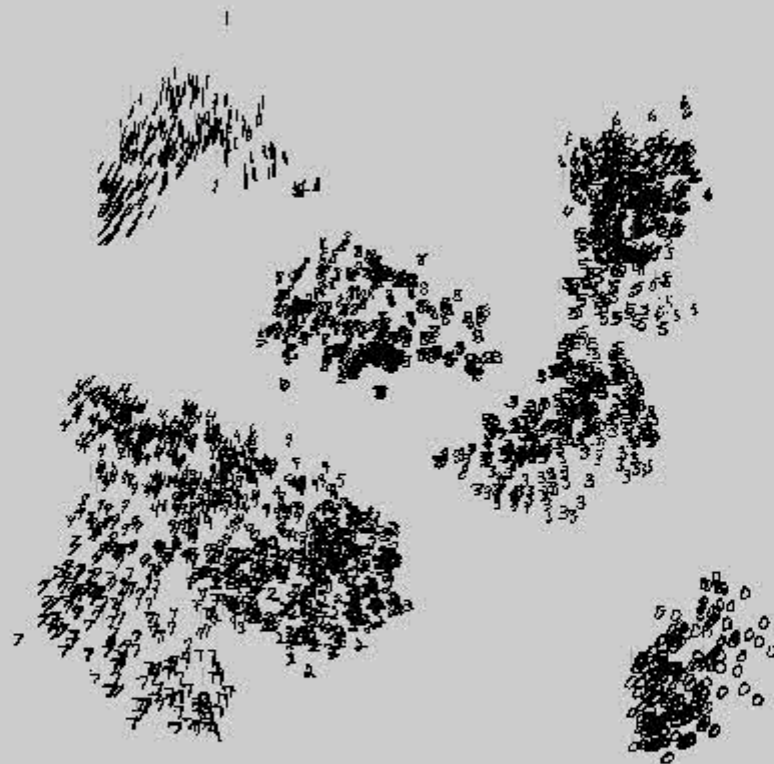
Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension
- ...





Source : <https://github.com/aferriss/mnist3d>



Mon quotidien en entreprise (en partie)



Développement de jeux vidéo mobiles



Mon quotidien en entreprise (en partie)



Développement de jeux vidéo mobiles

Coaching Equipe Data



Mon quotidien en entreprise (en partie)



Développement de jeux vidéo mobiles

Coaching Equipe Data

Entrainement de modèles de joueurs artificiels

Mon quotidien quotidien en entreprise (en partie)

Bazoo Block (jouer comme un joueur)



Mon quotidien quotidien en entreprise (en partie)

Belote, Coinche, Poker, Wordox ...

Apprendre à bien jouer en jouant





Quelques sources d'inspirations ...

Open AI

- <https://openai.com/blog/openai-five/>
- <https://openai.com/blog/better-language-models/>

DeepMind

- <https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/>

Nvidia

- <https://blogs.nvidia.com/blog/2019/03/18/gaugan-photorealistic-landscapes-nvidia-research/>

Two minutes papers

- <https://www.youtube.com/user/keeroyz>

Pour expérimenter soi même ...

Papers with Code

<https://paperswithcode.com/sota>

Microsoft ML Studio

<https://studio.azureml.net/>

Des Frameworks à n'en plus finir !

https://www.youtube.com/watch?v=SJldOOs4vB8&feature=youtu.be&bclid=IwAR1uDF-E_unybJDfkjdfACKiieQ2OA_SrEZrjYe_JclUuwltBTpJCPrhEbs







Pour expérimenter soi même ...

Papers with Code

<https://paperswithcode.com/sota>

Microsoft ML Studio

<https://studio.azureml.net/>

Des Frameworks à n'en plus finir !

https://www.youtube.com/watch?v=SJldOOs4vB8&=&feature=youtu.be&=&fclid=IwAR1uDF-E_unybJDfkjdFACKiieQ2OA_SrEZrjYe_JclUuwltBTpJCPrhEbs

Pour démarrer sur de bonnes bases ...

Le cours de Yaser S. Abu-Mostafa (CalTech)

- Learning From Data
- <https://work.caltech.edu/telecourse.html>





Merci de votre attention

Questions ?





Annexes





Annexes - thèse

Acknowledgments

Bharath is thankful to his PhD advisor for letting him work on this book during his nights and weekends, and especially thankful to his family for their unstinting support during the entire process.

Reza is thankful to the open source communities on which much of software and computer science is based. Open source software is one of the largest concentrations of human knowledge ever created, and this book would have been impossible without the entire community behind it.