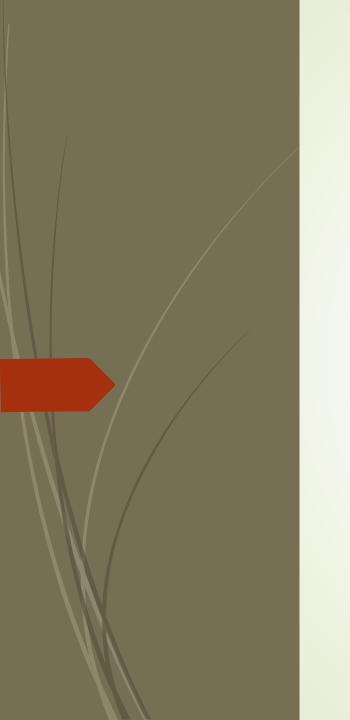
IA, Machine Learning, Deep Learning

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



Présentation

Présentation



Présentation - Formation





Présentation - Formation







Présentation – Enseignement



Présentation – Enseignement





Présentation – Enseignement





DPENCLASSROOMS



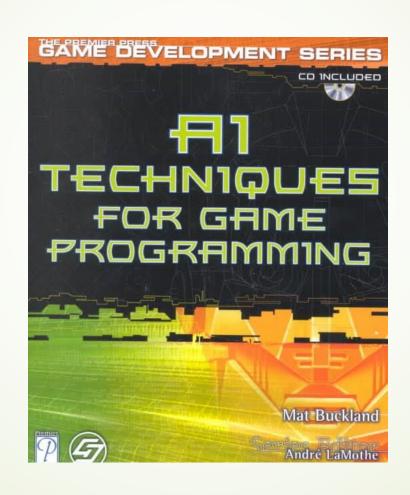


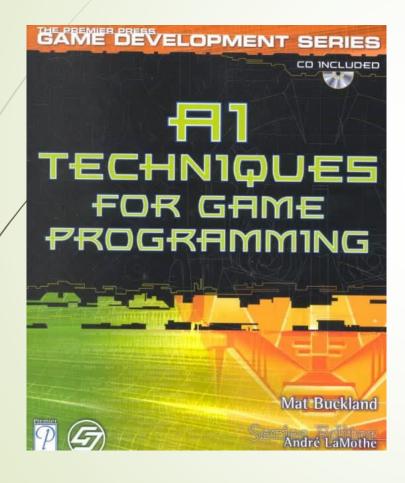


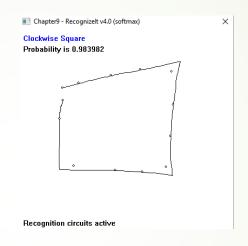


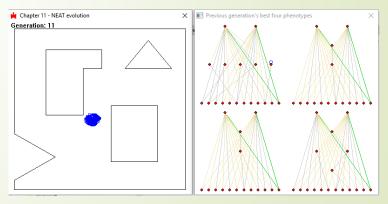


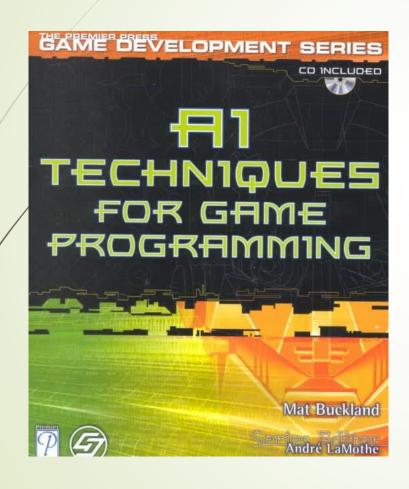


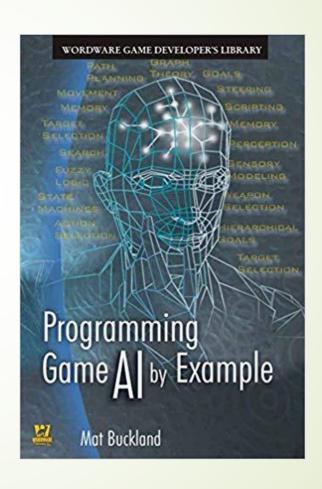


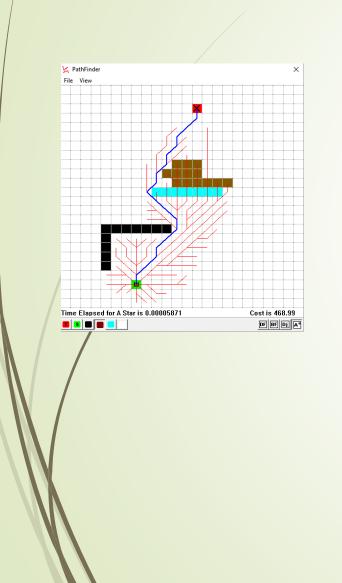


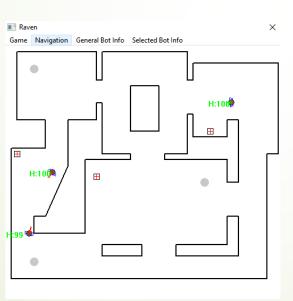


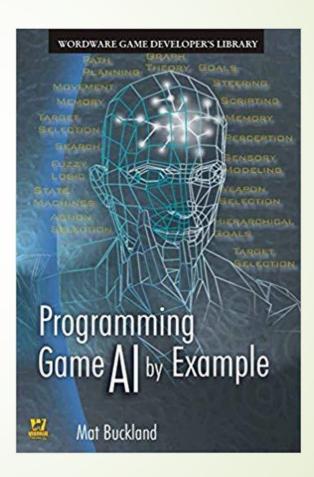




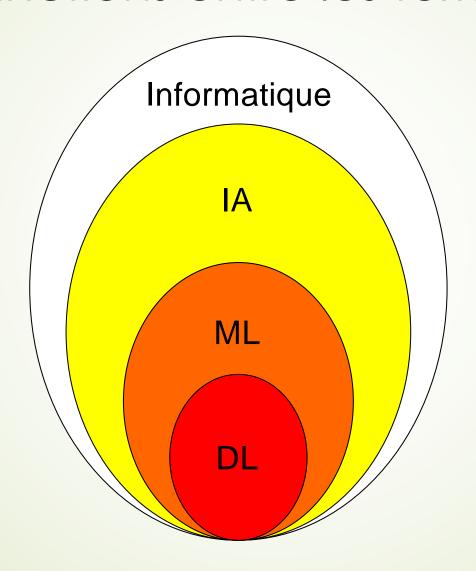


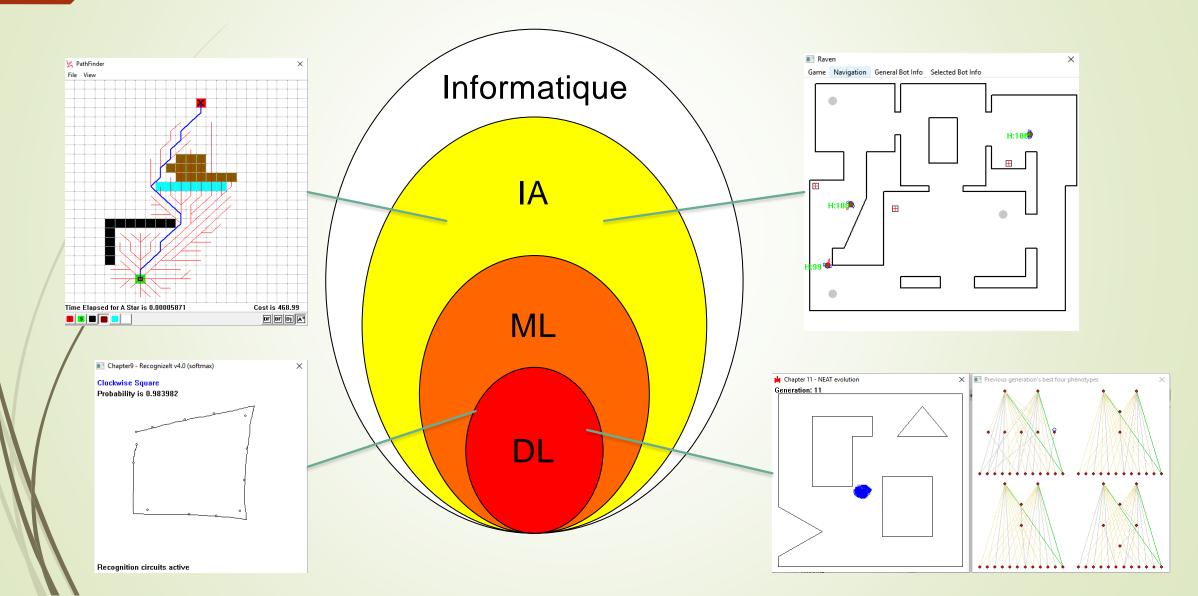


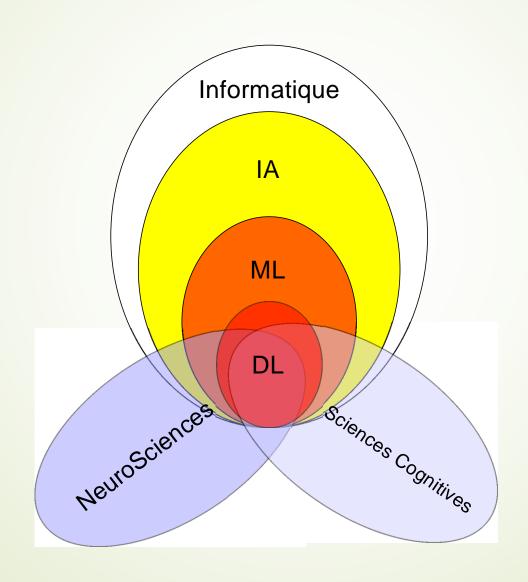




Terminologies







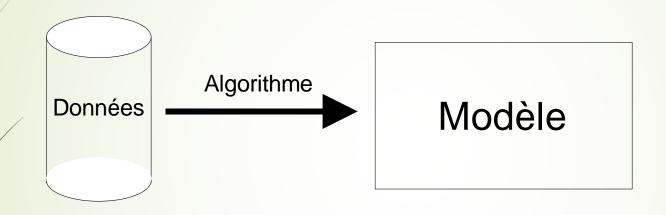
Intelligence Artificielle



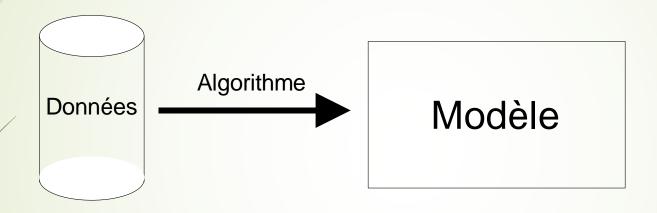
Intelligence Artificielle (M. Minsky)

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"

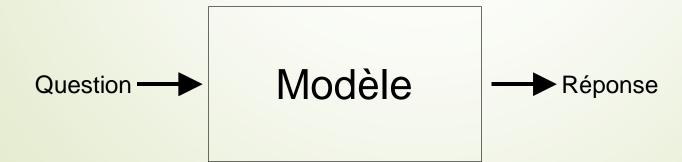
Apprentissage / Training



Apprentissage / Training



Utilisation / Inference



A quoi ressemble un modèle?

A quoi ressemble un modèle?

Pour être général il s'agit d'une fonction

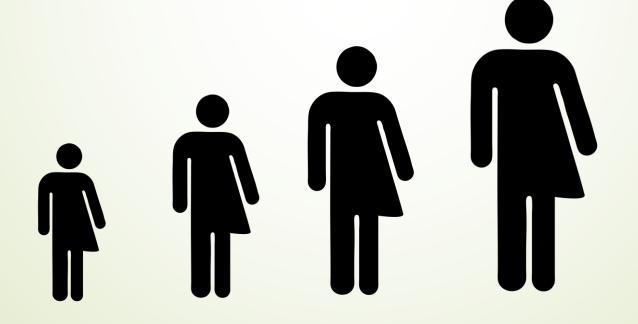
A quoi ressemble un modèle?

Pour être général il s'agit d'une fonction

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

Exemple:

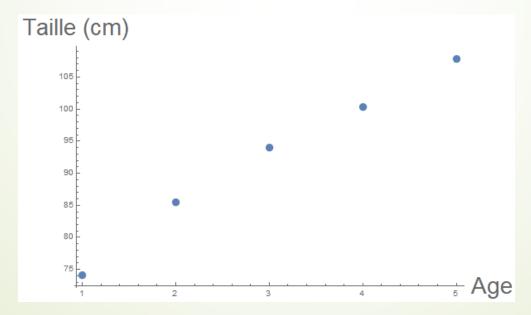
Problème jouet : trouver une fonction (un programme ?) qui soit capable de prédire la taille d'un humain en fonction de son âge...



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 :

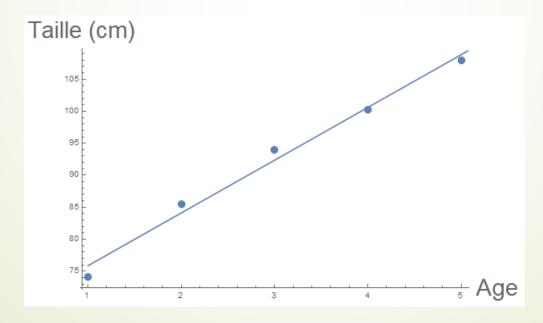


Source: https://www.disabled-world.com/calculators-charts/height-weight-teens.php

Exemple:

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

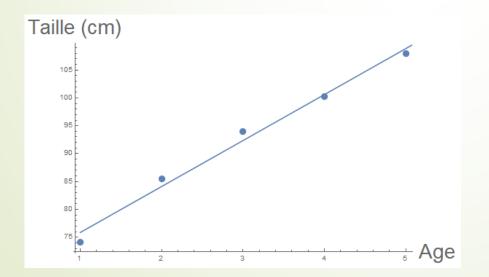
Modèle souhaité:



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:



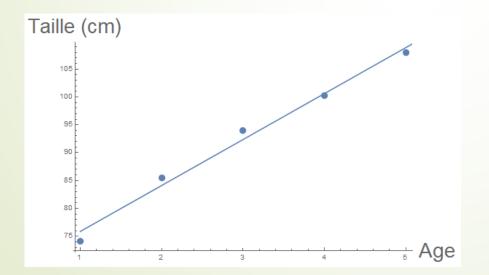
Modèle

Taille = $8.24 \times Age + 67.64$

Exemple:

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

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

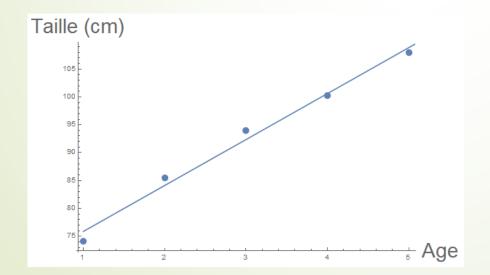


Modèle

Taille = $8.24 \times Age + 67.64$

Exemple:

Notre choix était-il judicieux ?



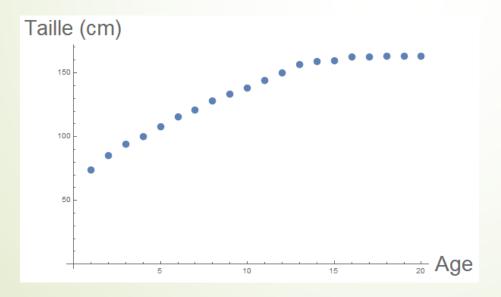
Modèle

Taille = $8.24 \times Age + 67.64$

Exemple:

Notre choix était-il judicieux ?

Données supplémentaires à disposition :

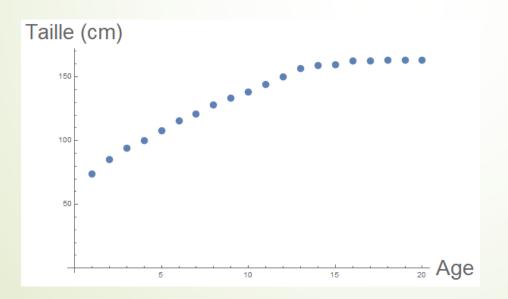


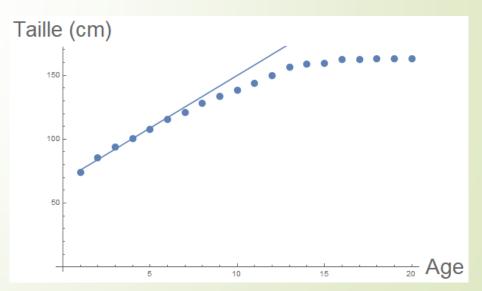
Source: https://www.disabled-world.com/calculators-charts/height-weight-teens.php

Exemple:

Notre choix était-il judicieux ?

Données supplémentaires à disposition incompatibles avec notre précédent modèle



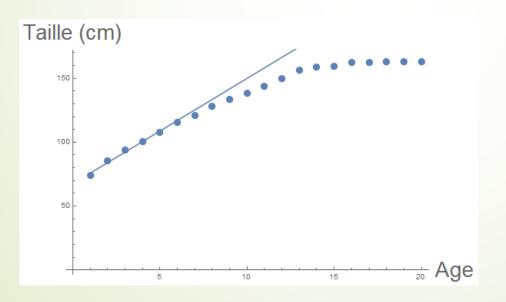


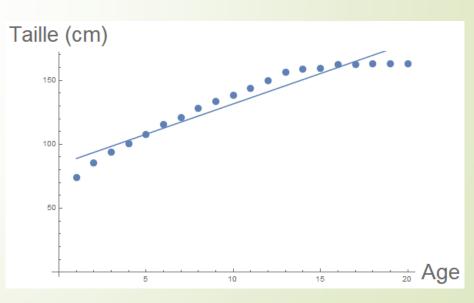
Source: https://www.disabled-world.com/calculators-charts/height-weight-teens.php

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

Comment détecter l'Underfitting?

Comment détecter l'Underfitting?

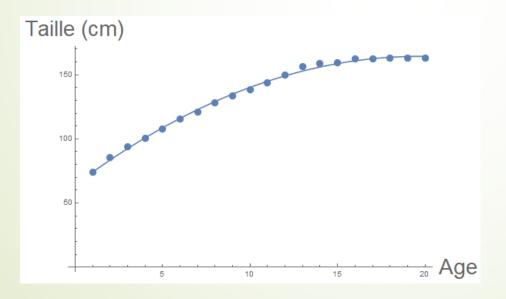
Monitoring de l'erreur sur la base d'exemple au cours de l'apprentissage

Erreur sur la base d'apprentissage Erreur en fin d'apprentissage trop importante Itérations (Epochs)

Exemple:

Notre choix était-il judicieux ?

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



Modèle

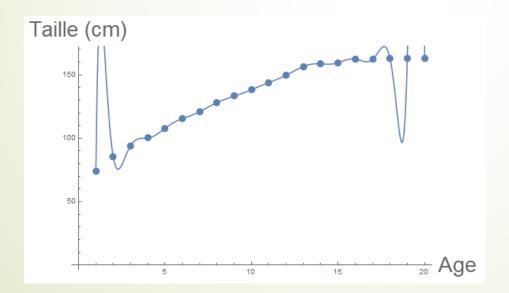
Taille = $-0.259 \times Age^2 + 10.19 \times Age + 64.25$

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

Exemple:

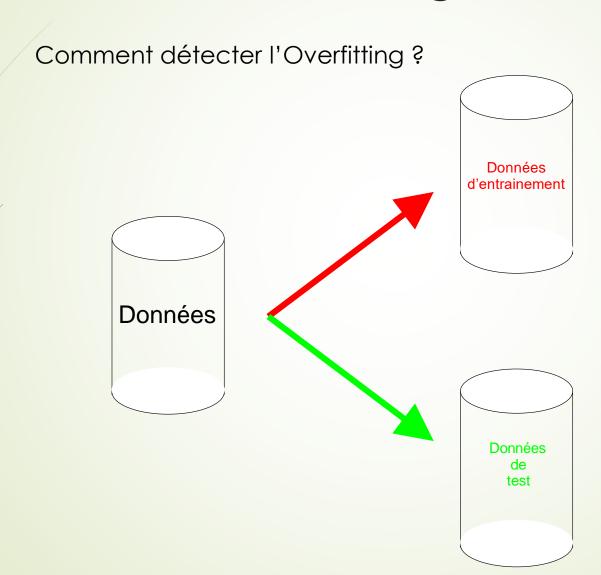
Attention, un modèle puissant (grand pouvoir d'expression) n'est pas forcément meilleur!

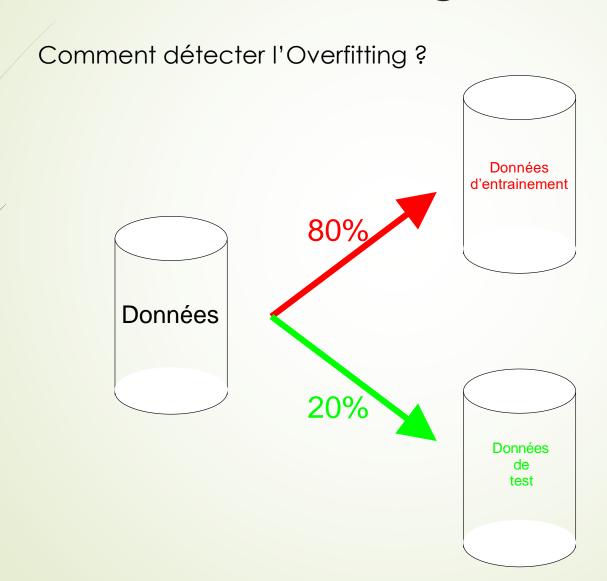
Nouveau choix de type de modèle : polynôme de degré 22 ! Cas de Surapprentissage (Overfitting)

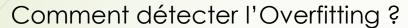


Modèle

Taille = -15109.6+48804.6 x Age-66501.1 x Age²+51711.5 x Age³-25882. x Age⁴+8869.49 x Age⁵-2149.14 x Age⁶+372.389 x Age⁷-45.6506 x Age⁸+3.77285 x Age⁹-0.178969 x Age¹⁰+0.00104904 x Age¹¹+0.000398215 x Age¹²-0.0000153565 x Age¹³-6.35198*10⁻⁷ x Age¹⁴+5.06753*10⁻⁸ x Age¹⁵+8.60159*10⁻¹⁰ x Age¹⁶-1.4655*10⁻¹⁰ x Age¹⁷+4.21198*10⁻¹³ x Age¹⁸+4.01457*10⁻¹³ x Age¹⁹-1.99247*10⁻¹⁴ x Age²⁰+4.15134*10⁻¹⁶ x Age²¹-3.36896*10⁻¹⁸ x Age²²

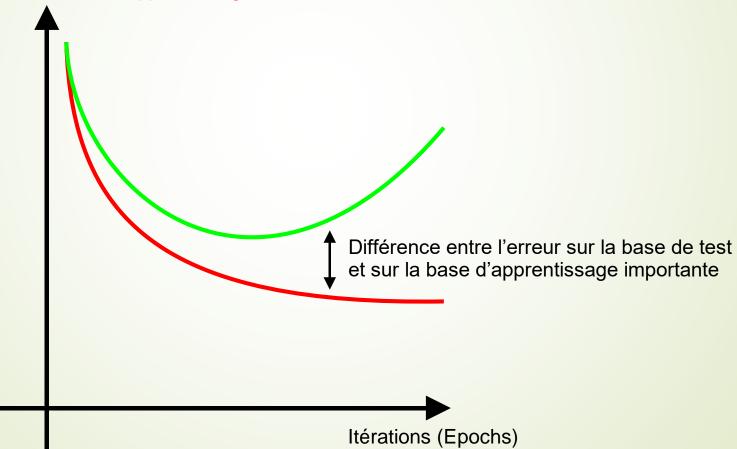






Erreur sur la base de test

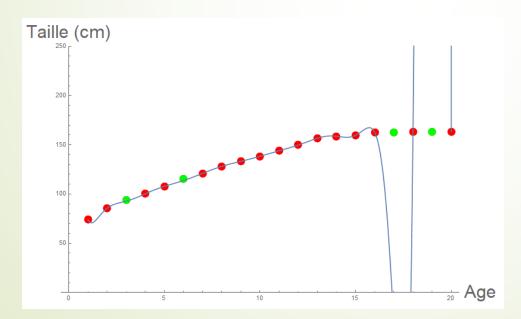
Erreur sur la base d'apprentissage



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)

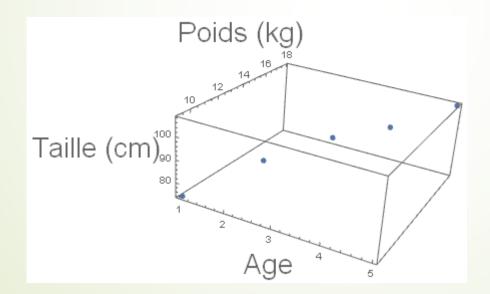


Modèle

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

Doit-on se limiter à une seule entrée ?

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

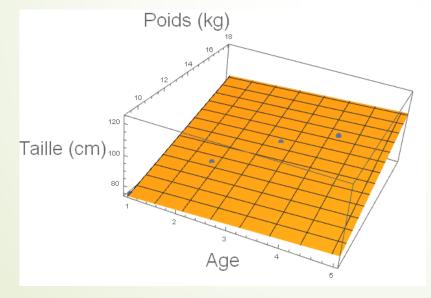


Source: https://www.disabled-world.com/calculators-charts/height-weight-teens.php

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

Taille = $1.74 \times Age + 3.13 \times Poids + 44.19$

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

Doit-on se limiter à une seule entrée ?

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

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

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

Doit-on se limiter à une seule entrée ?

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

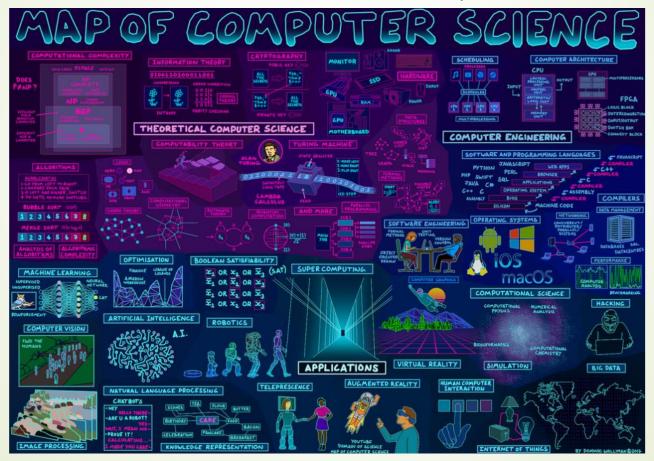
Et pourquoi pas : $y = ax + bz + cx^2 + dz^2 + exz + f$?

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

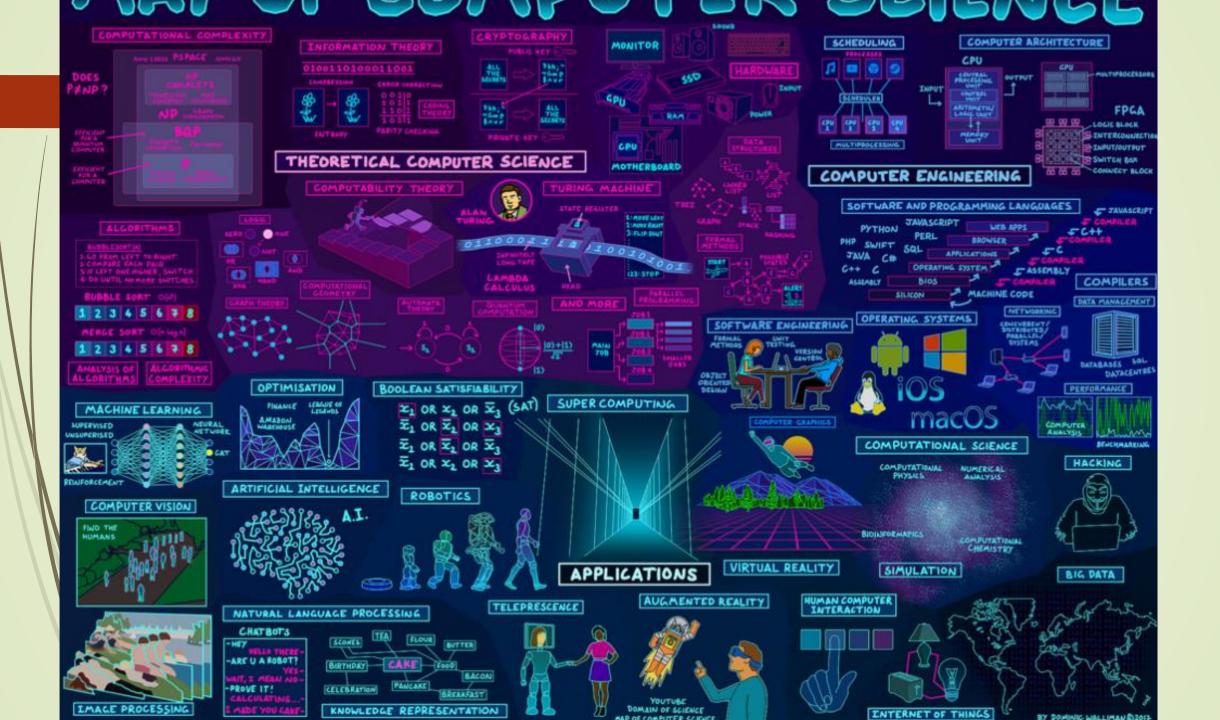
Ou encore: y = if x > b then c elif $z \le d$ then e else f?

Qui ici est déjà en overdose de notations mathématiques ?

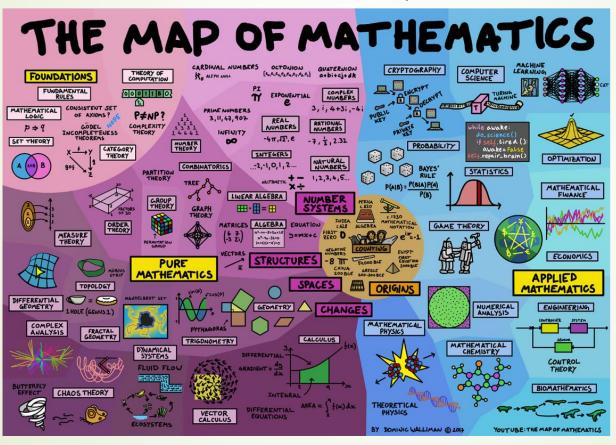
Quelques illustrations de nos disciplines ... (Dominic Walliman)



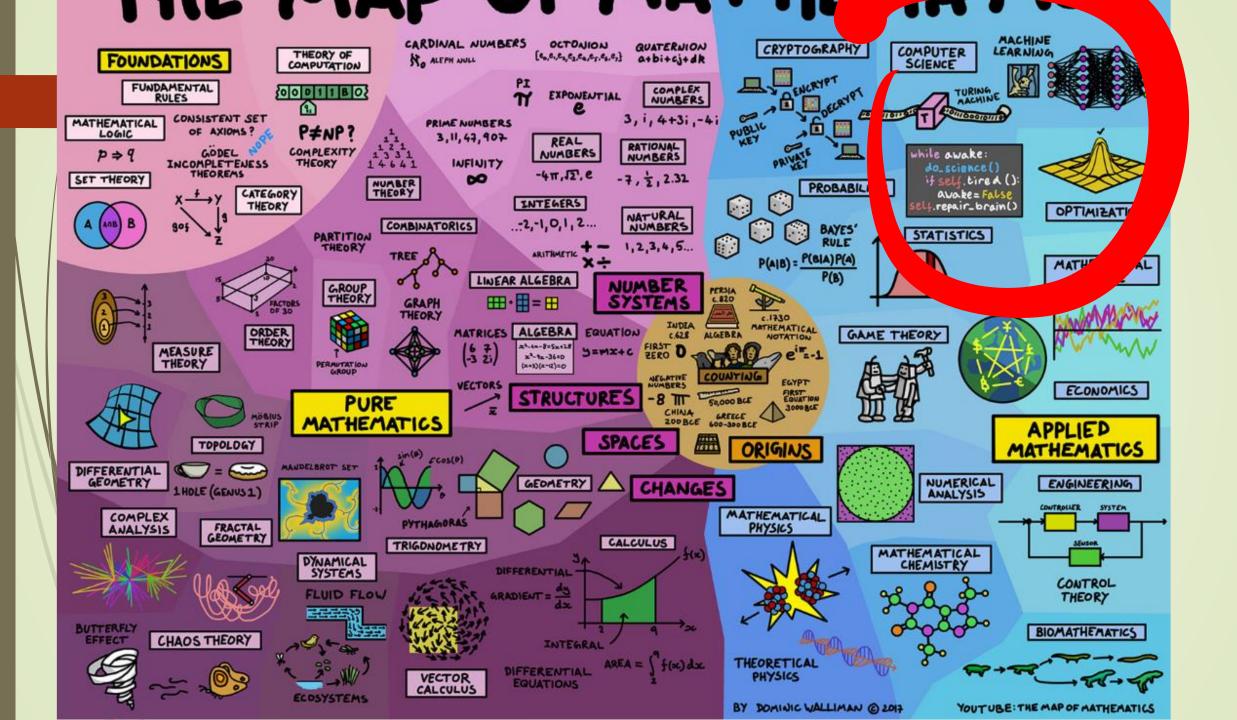
Source: https://www.flickr.com/photos/95869671@N08/



N'oubliez pas la vue d'ensemble ;)



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



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 = if x > b then c elif $z \le d$ then e else f?

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 a 1 couche cachée de neurones sigmoïdes qui approxime F a 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.

Théorème d'approximation universelle

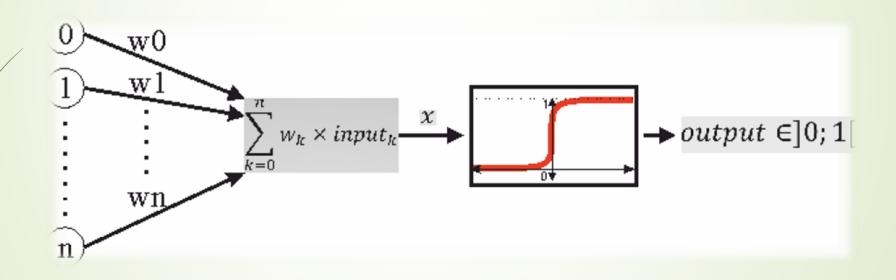
Cybenko 89

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

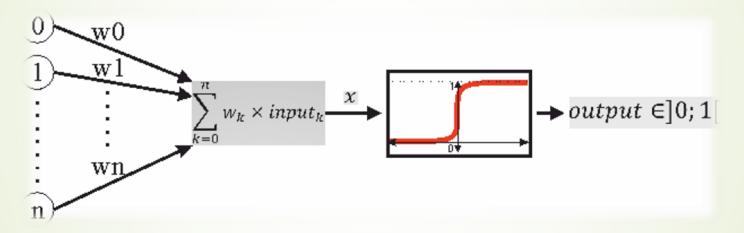
Sussman 92

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.

Neurone artificiel avec fonction d'activation sigmoïde

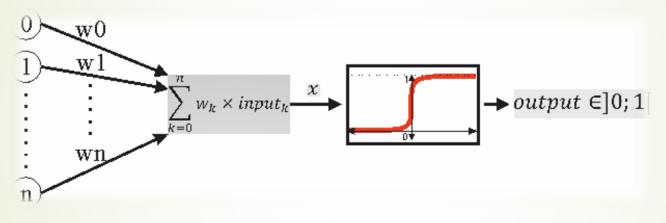


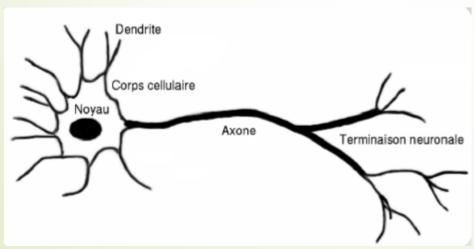
Neurone artificiel avec fonction d'activation sigmoïde



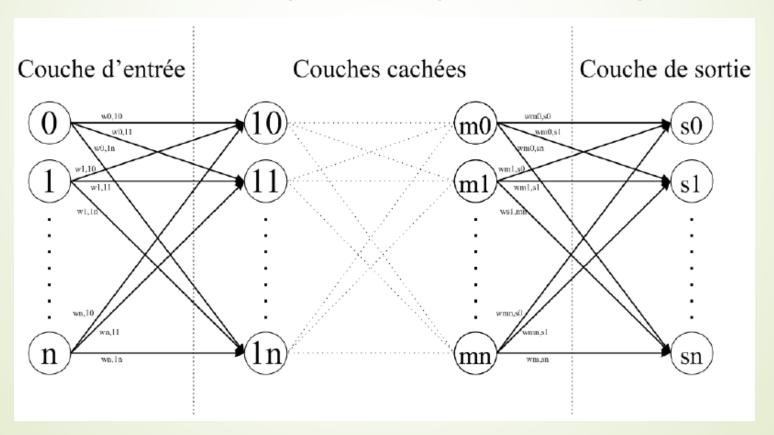
$$output = \frac{1}{1 + e^{-\sum_{k=0}^{n} w_k x_k}}$$

Métaphore Biologique





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



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

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

Démystification ...

Basic Math

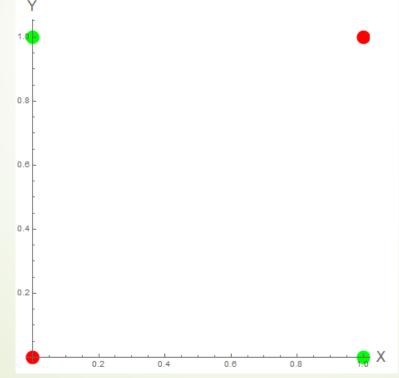
$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}$$

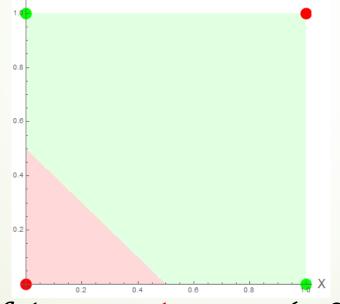
Dangerous Artificial Intelligence

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

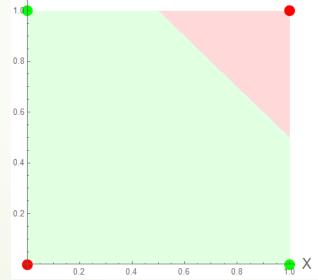
X	Y	XOR(X, Y)
/ True	True	False
True	False	True
False	True	True
False	False	False

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



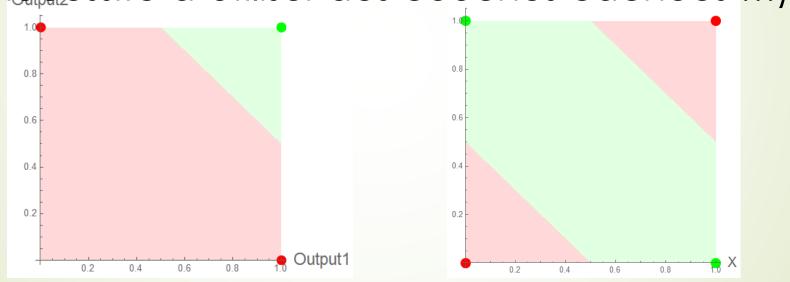


$$Output1 = if \ 1 \times x + 1 \times y + (-0,5) \ge 0 \ then \ 1 \ else \ 0$$
$$Output1 \approx sigmoid(1 \times x + 1 \times y + (-0,5))$$



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

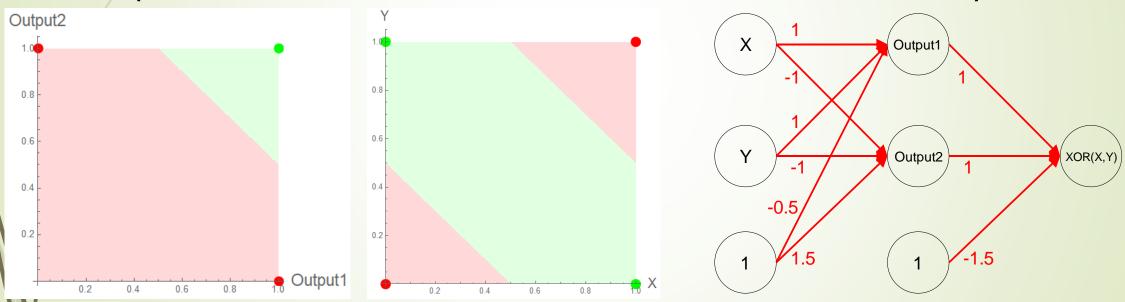
Exemple de problème de classification : le 'Ou' exclusif (XOR) (la nécessité d'utiliser des couches cachées ...)



$$XOR(x,y) = if \ 1 \times Output1 + 1 \times Output2 + (-1,5) \ge 0 \ then \ 1 \ else \ 0$$

 $XOR(x,y) \approx sigmoid(1 \times Output1 + 1 \times Output2 + (-1,5))$

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



$$XOR(x,y) = if \ 1 \times Output1 + 1 \times Output2 + (-1,5) \ge 0 \ then \ 1 \ else \ 0$$

 $XOR(x,y) \approx sigmoid(1 \times Output1 + 1 \times Output2 + (-1,5))$

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.

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'

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

Noisy Input Cell Feed Forward (FF) Radial Basis Network (RBF) Hidden Cell Probablistic Hidden Cell Spiking Hidden Cell Long / Short Term Memory (LSTM) Gated Recurrent Unit (GRU) Output Cell Match Input Output Cell Recurrent Cell Memory Cell Variational AE (VAE) Denoising AE (DAE) Different Memory Cell Kernel Convolution or Pool Hopfield Network (HN) Boltzmann Machine (BM) Restricted BM (RBM) Deep Convolutional Network (DCN) Deconvolutional Network (DN) Deep Convolutional Inverse Graphics Network (DCIGN) Generative Adversarial Network (GAN) Liquid State Machine (LSM) Extreme Learning Machine (ELM) Deep Residual Network (DRN) Kohonen Network (KN) Support Vector Machine (SVM) Neural Turing Machine (NTM)

A mostly complete chart of

Neural Networks

Backfed Input Cell
Input Cell

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

Distinctions ...

Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension
- -

Distinctions ...

Supervisé

- Classification
- Régression

Non Supervisé

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





Distinctions ...

Supervisé

- Classification
- Régression



Non Supervisé

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

Distinctions ...

Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension

- ...

Somme d'argent dépensée dans le magasin

Fréquence de visite dans le magasin

Distinctions ...

Supervisé

- Classification
- Régression

Non Supervisé

- Clustering
- Génération
- Réduction de dimension

- ...







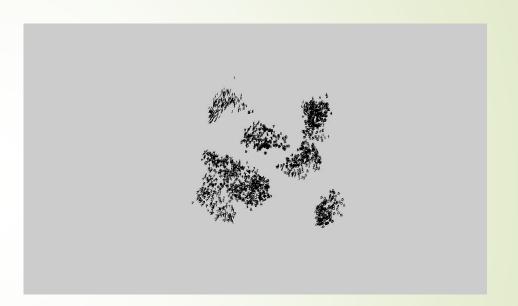
Distinctions ...

Supervisé

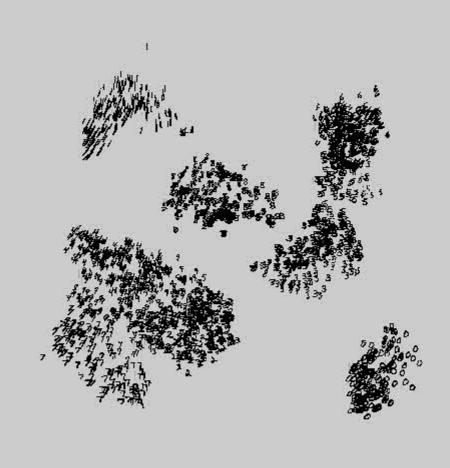
- Classification
- Régression

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 Al

- 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/

Tensorflow

https://www.tensorflow.org

pyTorch

https://pytorch.org/

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 - thèse

Acknowledgments 4 1 2

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.