

# IA, ML, DL

# *basics*

by JB

Le Mans School of AI - Saison 3 - 12/09/2019



# Intro

*“ L’IA est l’ensemble de technologies  
dont le but est de faire réaliser  
des tâches complexes à des machines.*

*— Olivier Bousquet*

*(directeur de recherches en Intelligence Artificielle chez Google)*



Concrètement il s'agit d'algorithmes complexes, qui vont apprendre à la machine par expérience.

**DATA**

**MODÈLE**  
**/**  
**PATTERN**

**PREDICTION**



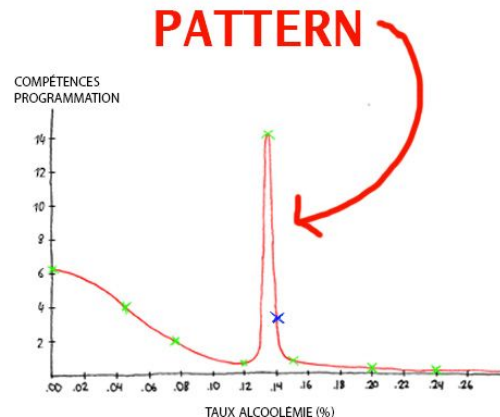
Exemple :

le “Ballmer Peak” de



**DATA**

COMPÉTENCES PROGRAMMATION	TAUX ALCOOLÉMIE (%)
6.24	0.00
4.00	0.0448
2.00	0.0776
0.72	0.12
14.24	0.1376
0.88	0.1496
0.40	0.20
0.32	0.24



**PRÉDICTION**

COMPÉTENCES PROGRAMMATION	TAUX ALCOOLÉMIE (%)
3.73	0.14

## UN APPRENTISSAGE EN DEUX PHASES

1

Phase d'entraînement

La machine utilise une partie  
des données pour apprendre

2

Phase de vérification

Et utilise l'autre pour valider ou  
invalider l'apprentissage

Datas  
brutes



*Collecte et  
référencement d'un  
grand nombre de  
données*

Organisation des  
datas



*Phase de  
traitement des  
données*

Construction  
d'un modèle

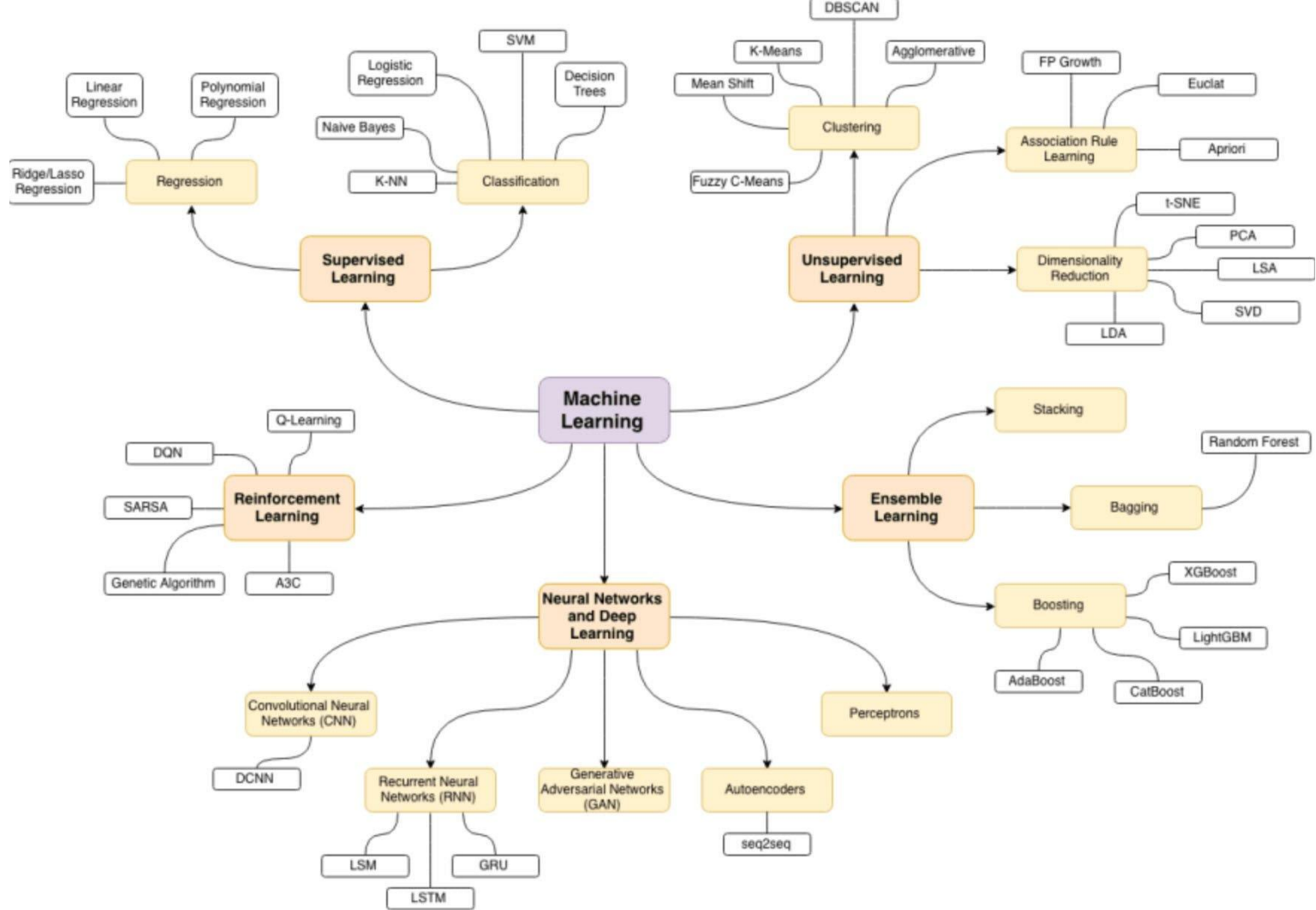


*L'algorithme est le  
modèle de "pensée"  
que la machine doit  
suivre*

Prédiction



*Analyse et  
prédit ce qui  
doit être fait*

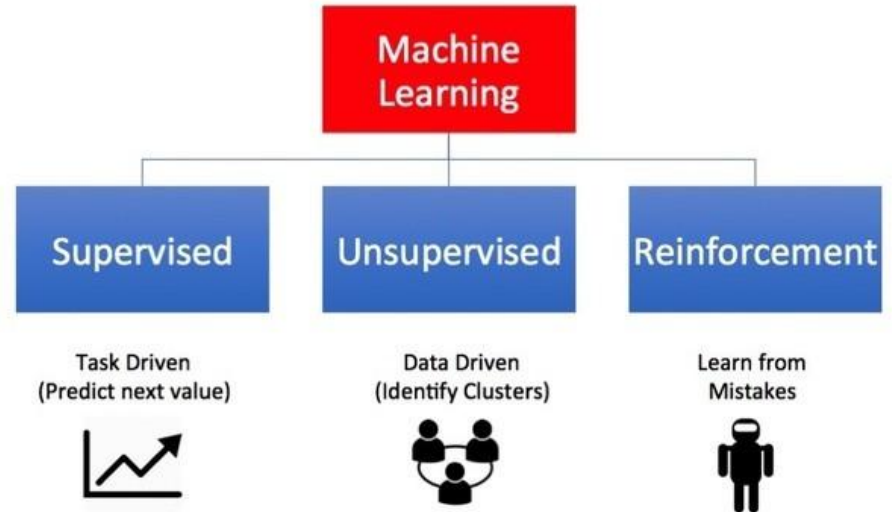




**L'enjeu :  
la formulation du problème  
à résoudre !**

# Machine Learning

## Types of Machine Learning





# Apprentissage Supervisé - *Supervised Learning*

avec des données annotées  
pour entraîner le modèle,

prédire la cible sur de nouvelles  
données non annotées.

airplane



automobile



bird



cat



deer



dog



frog



horse



ship

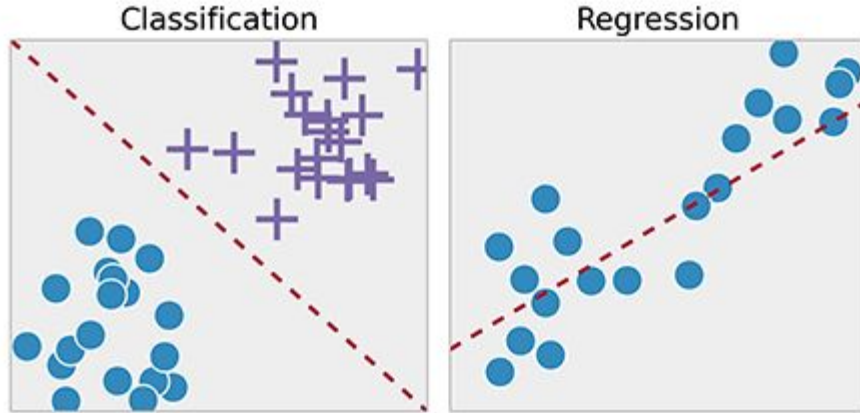


truck



# Régression et classification

autre distinction dans le choix d'un algorithme



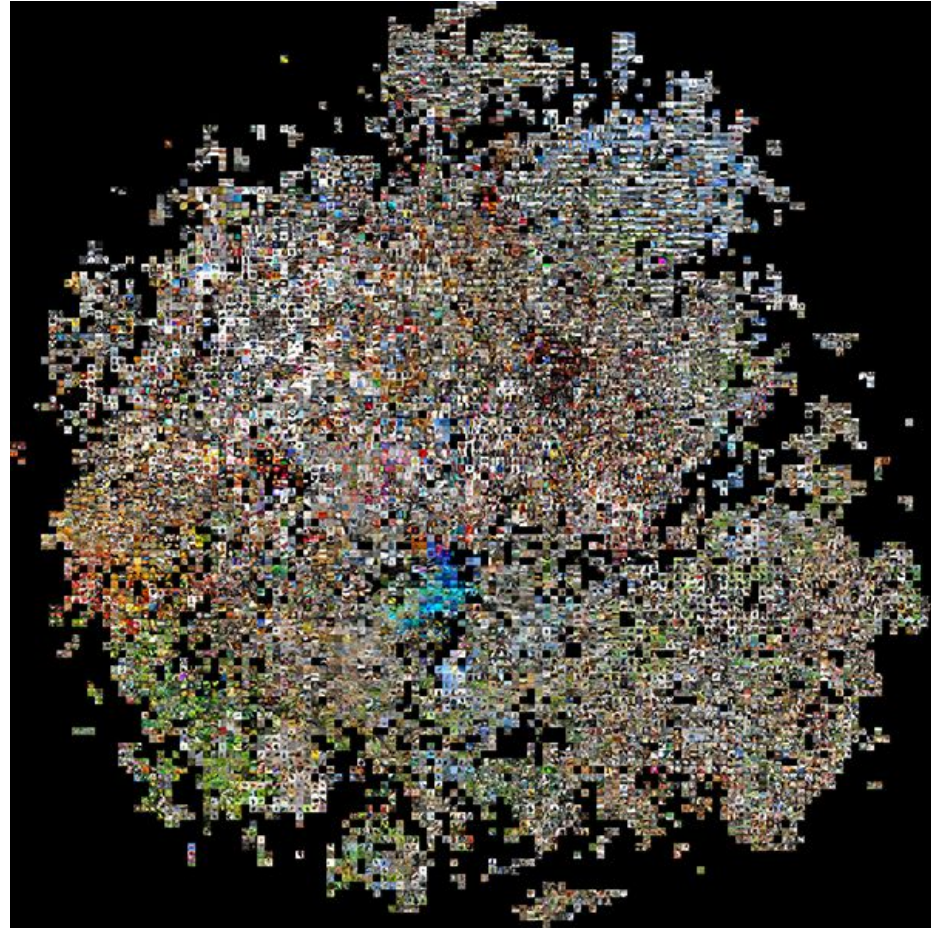
Quel est le **type de sortie** du programme :

- **est-ce une valeur continue (un nombre) ?**  
=> **une régression**
- **ou bien une valeur discrète (une catégorie) ?**  
=> **une classification.**

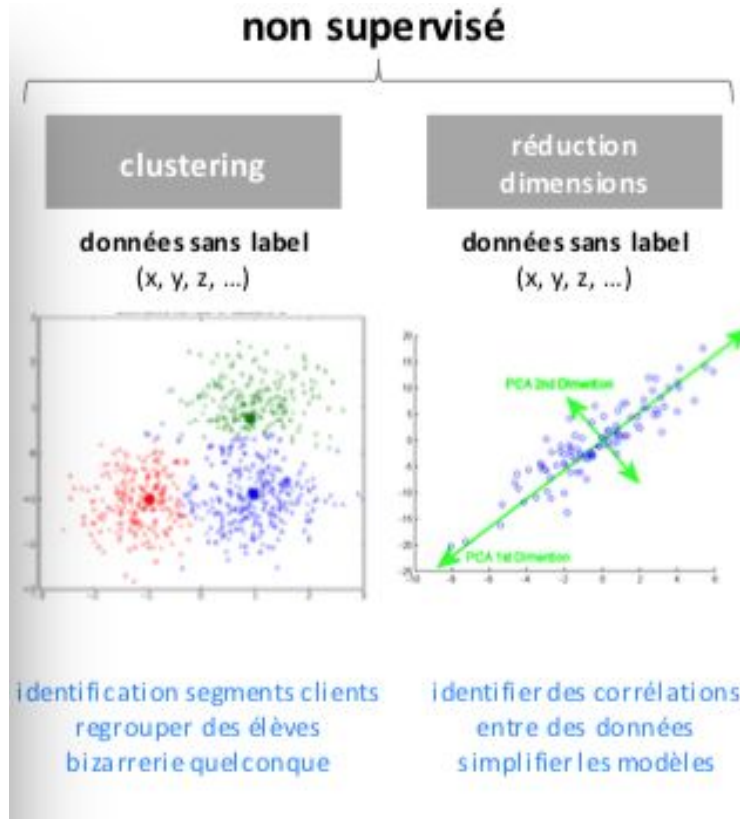
# Apprentissage non Supervisé - *Unsupervised Learning*

les données d'entrées ne sont pas  
annotées.

l'algorithme d'entraînement trouve  
des similarités regroupe ensemble  
celles qui partagent des  
caractéristiques communes.



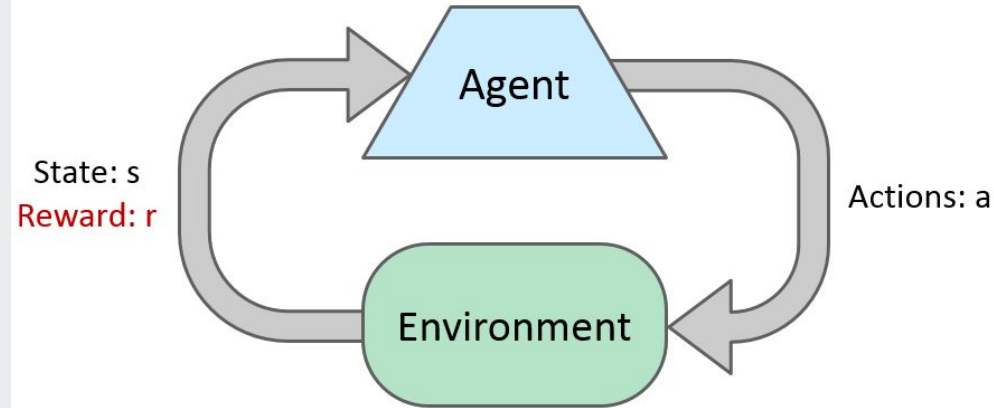
# Clustering et réduction de dimensions



- le terme « cluster » signifiant grappe/groupe (puisque l'on groupe les données)
- des k-plus proches voisins

# Apprentissage par renforcement - *Reinforcement Learning*

se base sur un cycle  
d'expérience / récompense et  
améliore les performances à  
chaque itération.



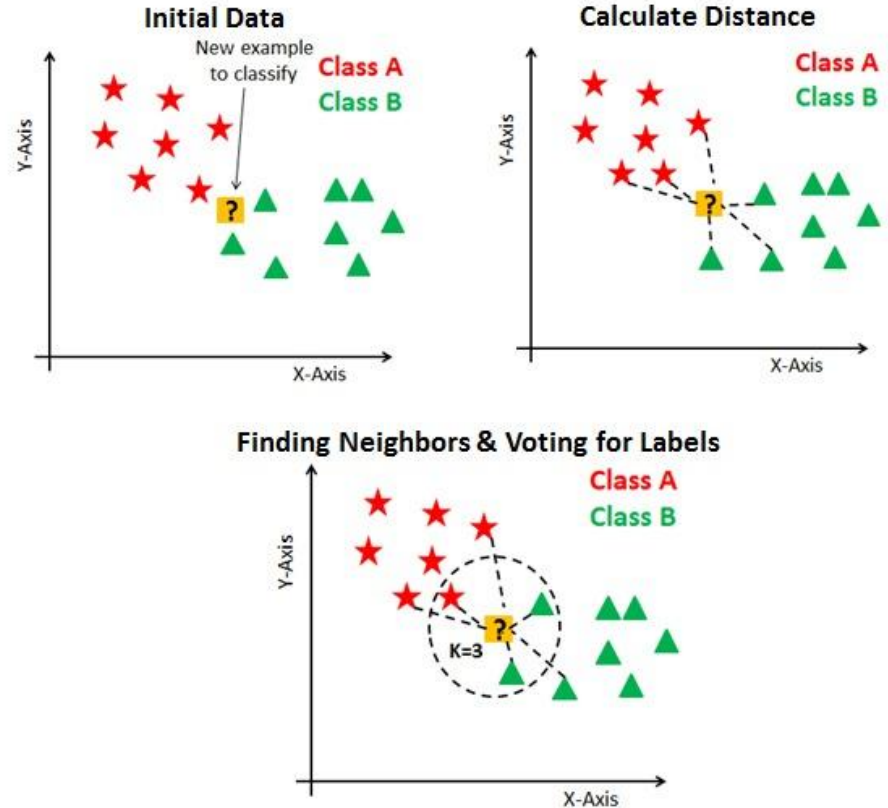


# Quelques algos célèbres de machine learning

# k-Nearest Neighbors (kNN)

– Supervisé – Classification

On souhaite, pour un nouveau produit, prédire à quelle famille il appartient grâce à ses caractéristiques, en identifiant les  $K$  voisins les plus proches.





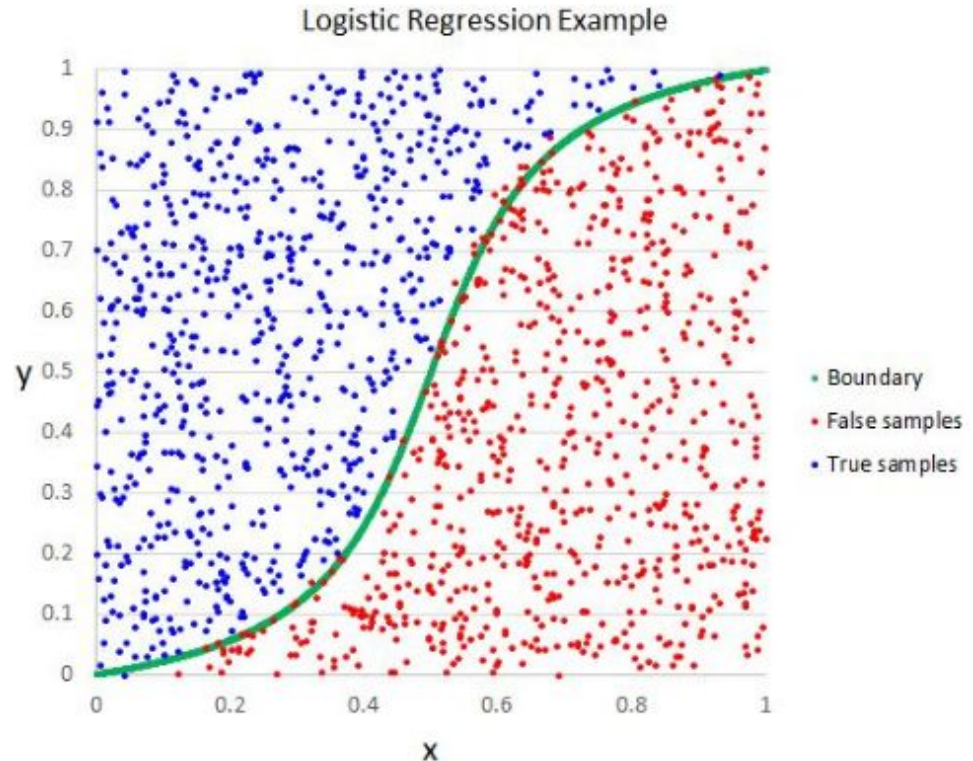
# Logistic Regression

## – Supervisé – Classification

La régression logistique est l'équation d'une courbe qui vient séparer les données en deux parties distinctes..

La régression logistique permettra de répondre à des problèmes comme :

- Est-ce que le client est solvable pour lui accorder un crédit ?
- Est-ce que la tumeur diagnostiquée est bénigne ou maline ?



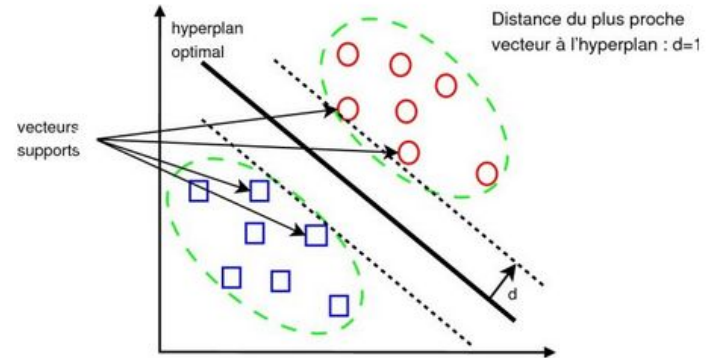
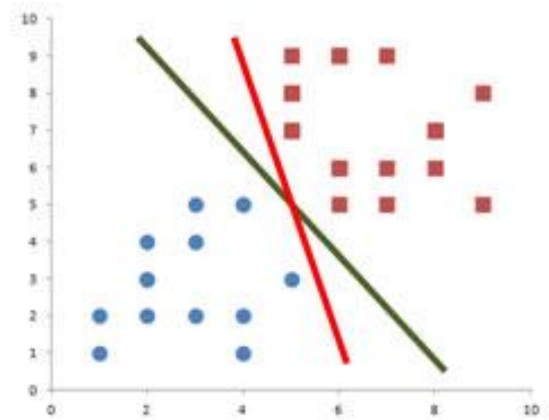


# Support Vector Machine (SVM)

– Supervisé – Régression ou Classification

La Machine à Vecteurs de Support (SVM) est lui aussi un algorithme de classification binaire.

Le SVM choisira la séparation la plus nette possible entre les deux classes (comme le trait vert). C'est pour cela qu'on le nomme aussi Large Margins classifier (classifieur aux marges larges).



La droite retenue (hyperplan) est celle dont la distance  $d$  (qui la sépare de tous les points au minimum) est maximale

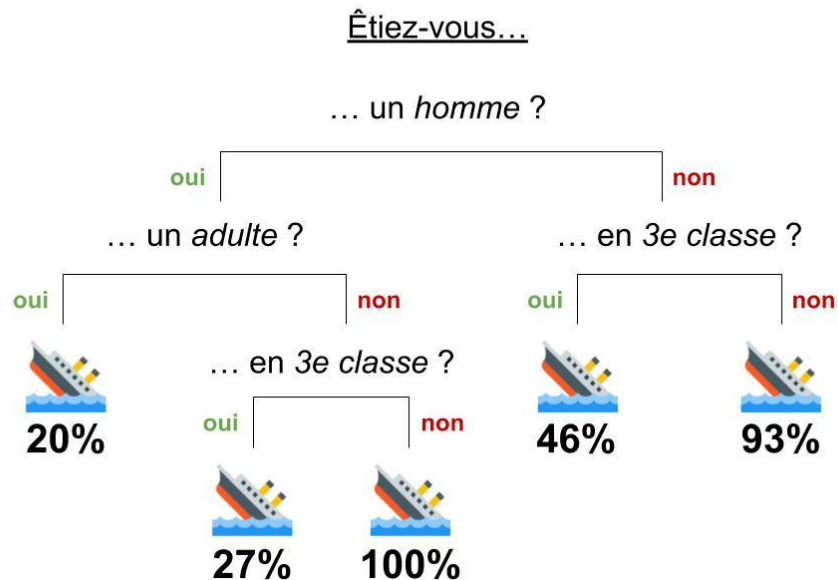
# Decision Trees

- *Supervisé – Régression ou Classification*

L'arbre de décision est un algorithme qui se base sur un modèle de graphe (les arbres) pour définir la décision finale.

Chaque nœud comporte une condition, et les branchements sont en fonction de cette condition (Vrai ou Faux). Plus on descend dans l'arbre, plus on cumule les conditions.

Quelles étaient vos chances de survie sur le Titanic ?  
(via un arbre de décision)

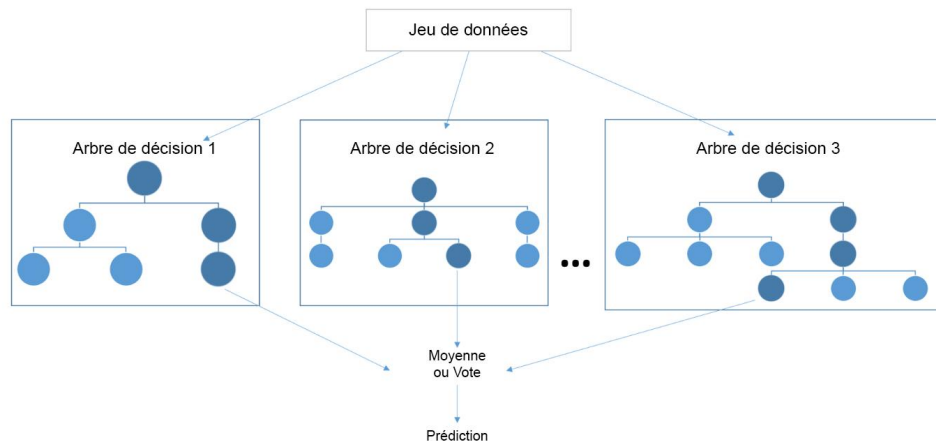


# Random Forest

- *Supervisé – Régression ou Classification*

L'arbre de décision est un algorithme qui se base sur un modèle de graphe (les arbres) pour définir la décision finale.

Chaque nœud comporte une condition, et les branchements sont en fonction de cette condition (Vrai ou Faux). Plus on descend dans l'arbre, plus on cumule les conditions.

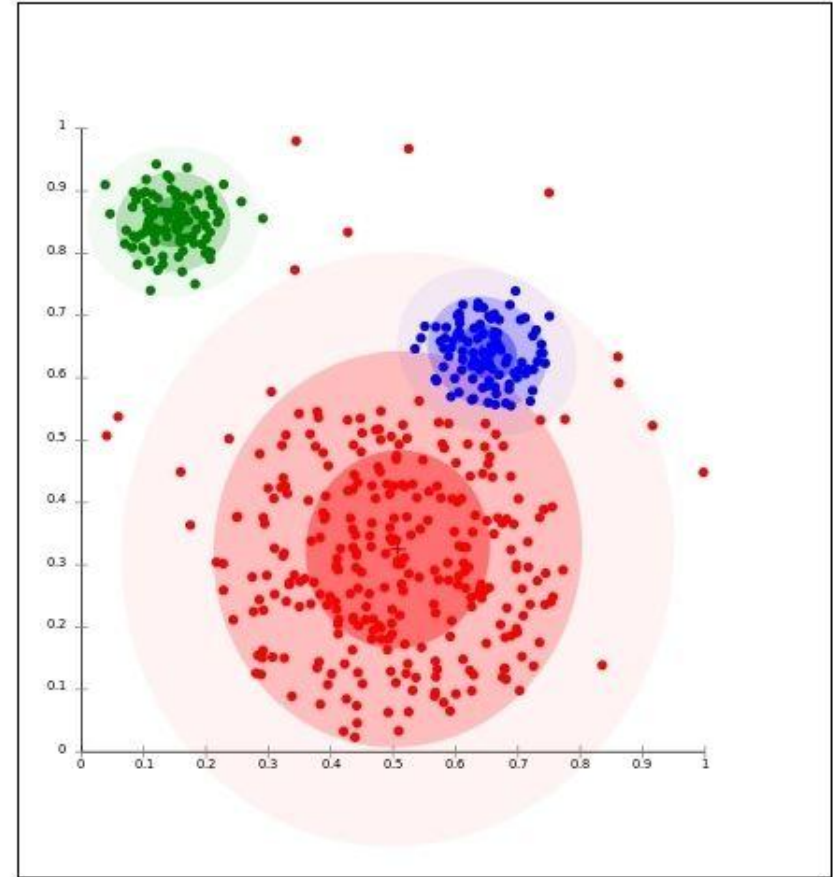


# K-Means

– *Non-supervisé - clustering*

On lui donne un ensemble d'éléments (des données), et un nombre de groupes K.

K-means va segmenter en K groupes les éléments. Le groupement s'effectue en minimisant la distance euclidienne entre le centre du cluster et un élément donné.





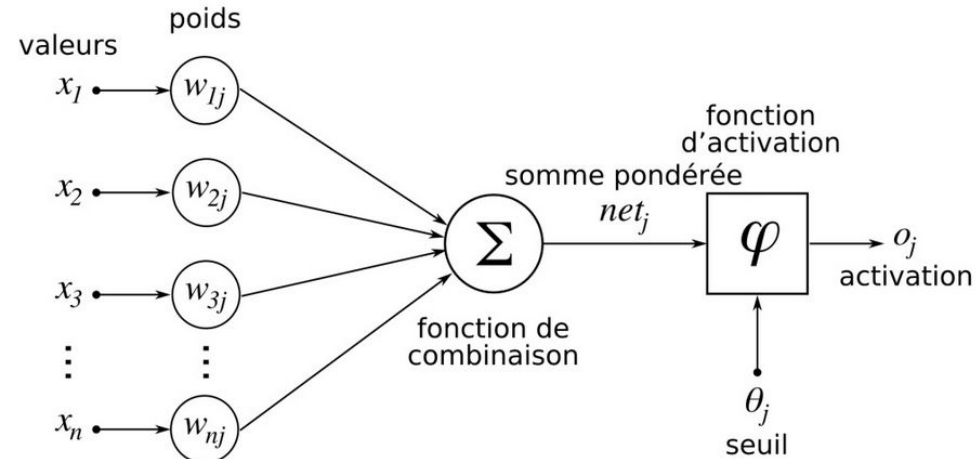
# Deep Learning et réseaux de neurones

# Neurone artificiel

il y a plusieurs variables auxquels sont affectés des coefficients (ou poids).

Via une fonction de combinaison on obtient une somme de ces variables en prenant en compte les coefficients.

Cette somme passe enfin dans une fonction d'activation qui va décider de la valeur de sortie du neurone. Cette fonction d'activation va utiliser un seuil pour faire son calcul. Si la somme des variables est en dessous du seuil le neurone est dit non-actif et sa sortie vaut 0 ou -1. Si au contraire la somme est au dessus alors le neurone est dit actif et sa sortie vaut 1.



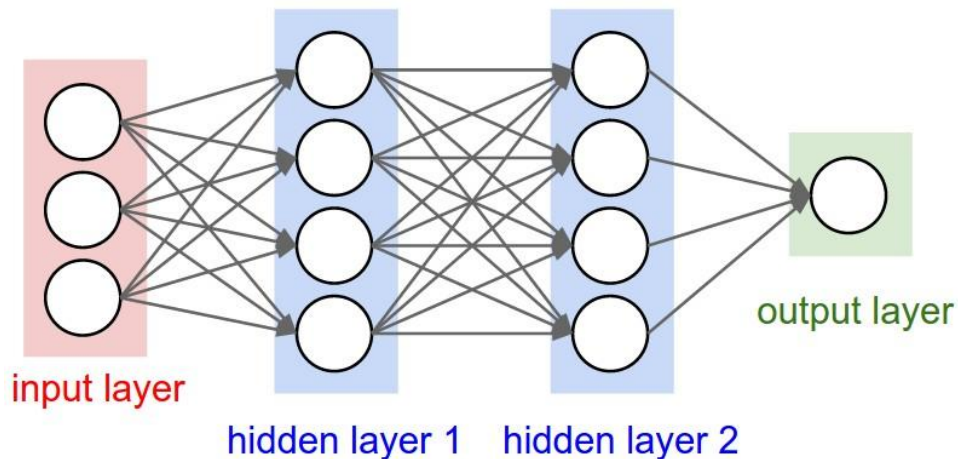
# Réseau de neurones

Dans ces réseaux,

on retrouve le tiers d'entrée (Input Layer) qui va recevoir les données d'entrées.

L'Input Layer va propager les données par la suite aux tiers cachés (Hidden Layers).

Finalement le Tiers de sortie (le plus à droite) permet de produire le résultat de classification.



# ALGORITHMES DE DEEP LEARNING

Source : [www.penseeartificielle.fr](http://www.penseeartificielle.fr)

## ★ RÉSEAU DE NEURONES ARTIFICIELS (ANN)

*Simple mais sert à tout*

**Prédictions** ⊖ Sans notion de temporalité  
Analyse d'image (peu efficace) ⊖ Classification d'objets

## ★ RÉSEAU DE NEURONES CONVOLUTIFS (CNN)

*Très utilisé*

**Analyse d'image** ⊖  
1 Imagerie médicale  
2 Reconnaissance d'objets  
3 Reconnaissance faciale  
Analyse de texte, de son...

## ★ RÉSEAU ANTAGONISTE GÉNÉRATIF (GAN)

*Très utilisé et puissant*

**Génération d'image** ⊖  
1 Visage  
2 Texture  
3 Tableau  
Génération de jeux de données  
Génération de texte, de voix

## ★ RÉSEAU DE NEURONES RÉCURRENTS (RNN)

*Fortement utilisé dans sa version "bi-LSTM"*

Prédictions temporelles ⊖ Maintenance prédictive  
Prévisions de stocks  
**Langage naturel** ⊖ Analyse de texte  
Synthèse, génération de texte

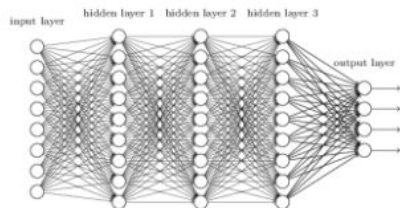
## ★ APPRENTISSAGE PAR RENFORCEMENT

*Prometteur*  
**Robotique**

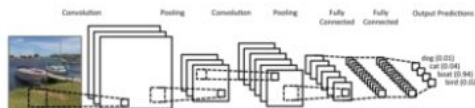
## ★ AUTRES

Carte auto-adaptative  
Machine de Boltzmann restreinte  
Auto-encoders  
...

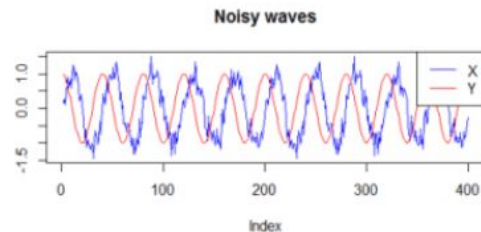




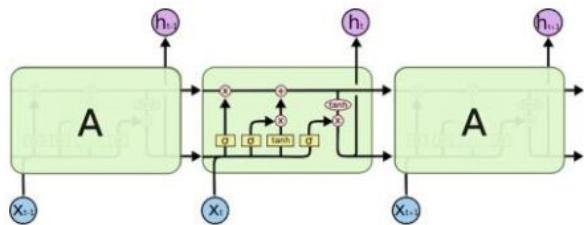
**fully connected**  
classification  
et prédictions



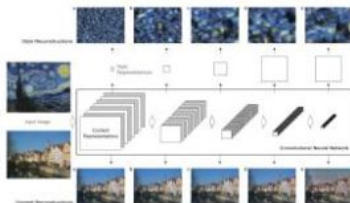
**convolutionnels**  
spatial  
*reconnaissance images*



**récurrents**  
temporels  
*ECG, finance, bruit*



**LSTM**  
contexte - bidirectionnel  
*traduction, dialogue, recherche*



**transfer networks**  
apprentissage incrémental  
*changement de domaine*



**génératifs**  
variations – augmentation  
*modification d'images et de textes*

A mostly complete chart of

# Neural Networks

©2016 Fjodor van Veen - asimovinstitute.org

-  Backfed Input Cell
-  Input Cell
-  Noisy Input Cell
-  Hidden Cell
-  Probabilistic Hidden Cell
-  Spiking Hidden Cell
-  Output Cell
-  Match Input Output Cell
-  Recurrent Cell
-  Memory Cell
-  Different Memory Cell
-  Kernel
-  Convolution or Pool

Perceptron (P)



Feed Forward (FF)



Radial Basis Network (RBF)



Deep Feed Forward (DFF)



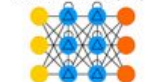
Recurrent Neural Network (RNN)



Long / Short Term Memory (LSTM)



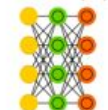
Gated Recurrent Unit (GRU)



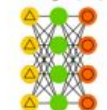
Auto Encoder (AE)



Variational AE (VAE)



Denoising AE (DAE)



Sparse AE (SAE)



Markov Chain (MC)



Hopfield Network (HN)



Boltzmann Machine (BM)



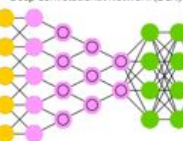
Restricted BM (RBM)



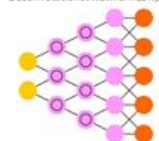
Deep Belief Network (DBN)



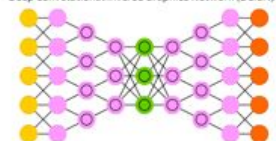
Deep Convolutional Network (DCN)



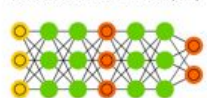
Deconvolutional Network (DN)



Deep Convolutional Inverse Graphics Network (DCIGN)



Generative Adversarial Network (GAN)



Liquid State Machine (LSM)



Extreme Learning Machine (ELM)



Echo State Network (ESN)



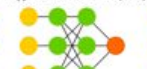
Deep Residual Network (DRN)



Kohonen Network (KN)



Support Vector Machine (SVM)



Neural Turing Machine (NTM)

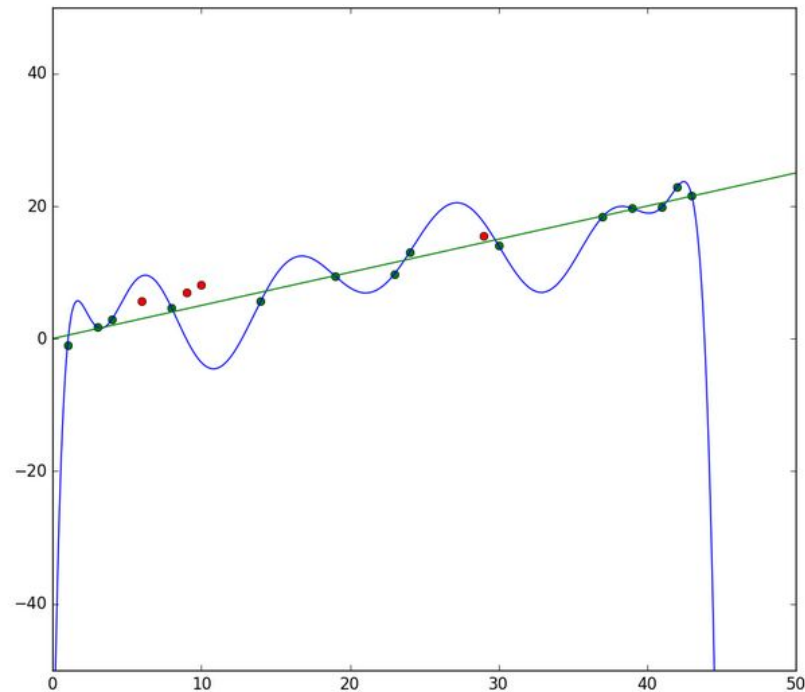


# sur-apprentissage

- *overfitting*

Un modèle trop spécialisé sur les données du Training Set et qui se généralisera mal.

=> l'équilibre est très difficile à trouver (combien de couches, combien de neurones, quels autres paramètres choisir aussi) et demande simplement de l'expérience, du test, et de l'analyse.



L'image ci-dessus montre un exemple d'Overfitting. Le tracé en bleu représente une fonction de prédiction qui passe par toutes les données du *Training Set* (points en vert). On voit bien que la fonction est instable (grande variance) et qu'elle s'écarte beaucoup des points rouges qui représentent des données non vues lors de la phase d'apprentissage (*Test Set*).

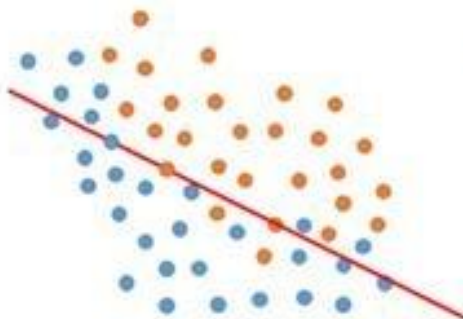
# sous-apprentissage

- *underfitting*

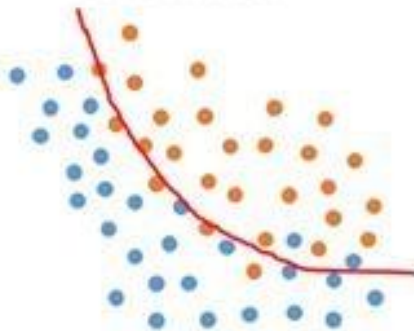
un modèle généraliste incapable de fournir des prédictions précises.

Autrement dit, le modèle prédictif n'arrive même pas à capturer les corrélations du Training Set.

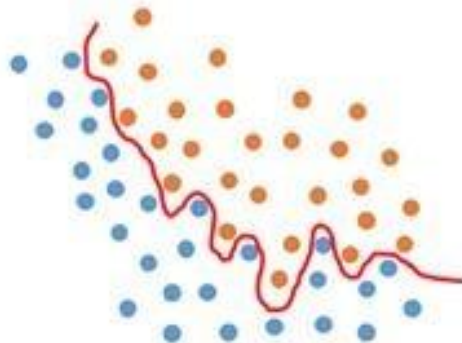
Sous-apprentissage



Bon modèle



Sur-apprentissage

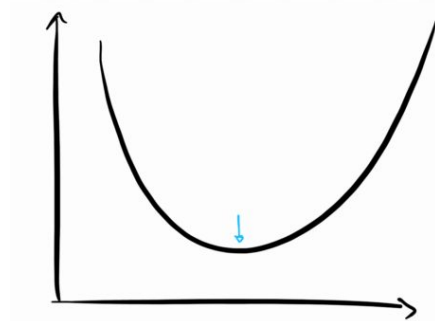


# Descente de gradient

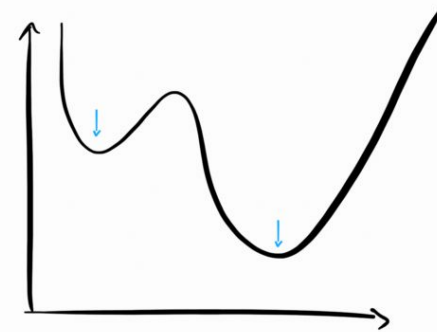
La Descente de Gradient est un algorithme d'optimisation qui permet de trouver le minimum de n'importe quelle fonction convexe, c'est-à-dire une fonction qui a l'allure d'une belle vallée qui descend progressivement vers un unique minimum.

l'algorithme est utilisé pour minimiser la fonction coût, qui justement est une fonction convexe

=> l'algorithme de Gradient Descent est fondamental: La machine apprend grâce à lui.

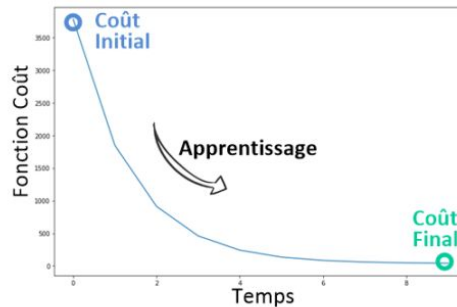


Fonction Convexe

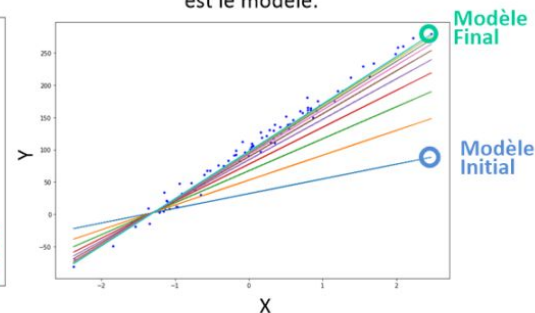


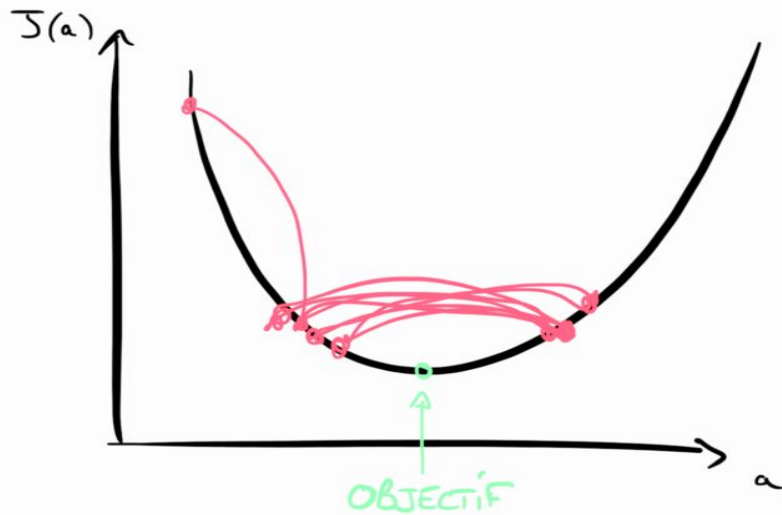
Fonction Non-Convexe

Minimisation de la Fonction Coût



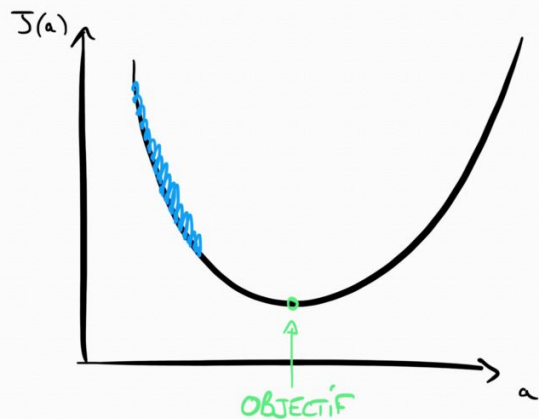
Plus la Fonction Coût est faible, meilleur est le modèle.





# Vitesse d'apprentissage

- *learning rate*



C'est un hyper paramètre (ce n'est pas un paramètre du modèle mais qui influe sur les performances du modèles).

c'est le pas dans la descente de gradient.

# Matrice de confusion

Une Confusion Matrix est un résumé des résultats de prédictions sur un problème de classification.

=> permet de mesurer la performance d'un modèle.

The Confusion Matrix

PREDICTED \ ACTUAL	POSITIVE	NEGATIVE
Positive	TRUE POSITIVE	FALSE POSITIVE Type I Error
Negative	FALSE NEGATIVE Type II Error	TRUE NEGATIVE

# Autres termes

- batch : lors des phases d'apprentissage, il est judicieux d'envoyer les données par batch avant de procéder à la mise à jour du réseau pour l'améliorer
- dropout : sujet qui fait débat en ce moment, car un brevet a été déposé sur le concept par Google. Le principe est de jeter une partie aléatoire de l'information circulant dans un réseau pour le forcer à s'adapter



K-means : <https://mrmint.fr/kmeans-python-scikit-learn>

Réseau de neurones :

<https://superdatabros.wordpress.com/2018/04/04/computer-vision-deep-learning-tomates-mures/>