Saagie©

Deep Learning avec Pytorch

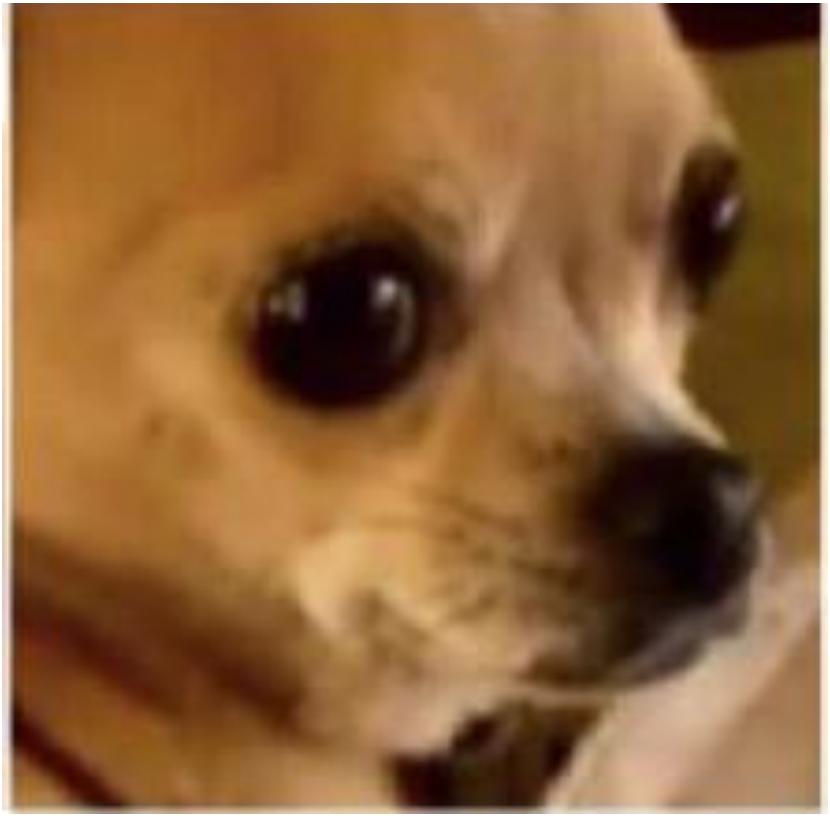
Chihuahua

Saagie

Muffin

Saagie

Chihuahua



Saagie

Chihuahua

Saagie

Muffin

Saagie

Chihuahua

Saagie

Muffin

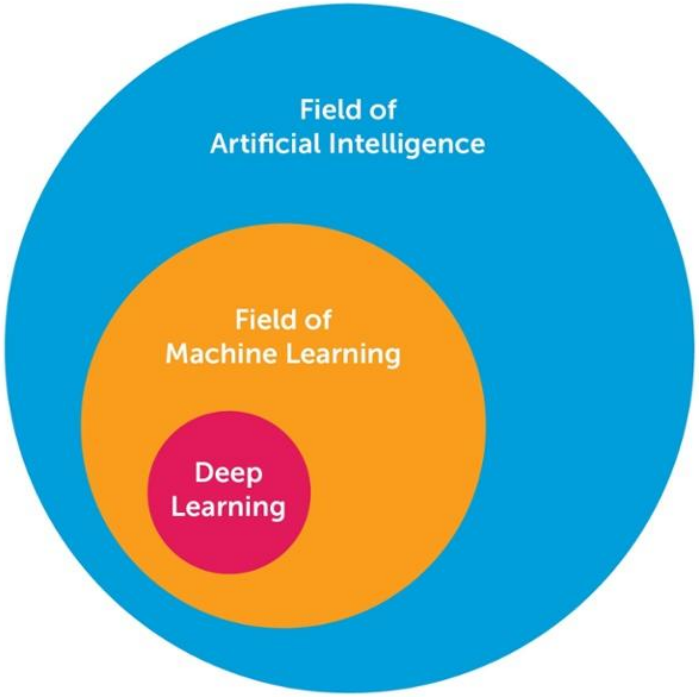
Saagie

Chihuahua vs Muffin

Est ce possible de classifier **automatiquement** ces images ?

Saagie

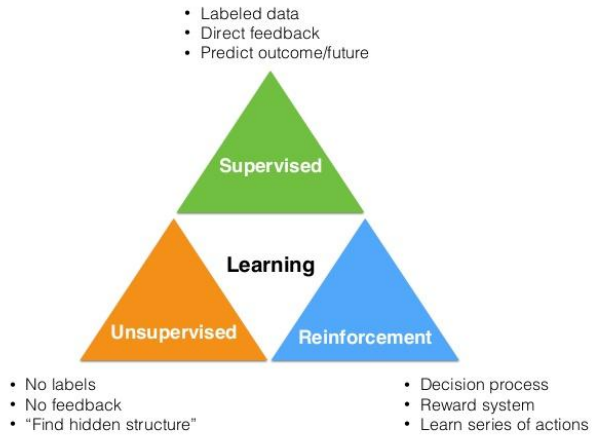
Deep Learning



Saagie

Machine Learning

Machine Learning : Donner la possibilité à une machine d'apprendre sans avoir été programmée explicitement



+ Apprendre à modeliser un problème

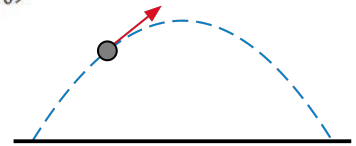
Trajectory Equation:

$$y - y_0 = (\tan\theta)(x - x_0) - \frac{g}{2v_0^2}(1 + \tan^2\theta)(x - x_0)^2$$

θ – launch angle

v_0 – launch velocity

x_0, y_0 – launch position, positive up



+ Apprentissage de la modélisation à partir de données

+ Classification

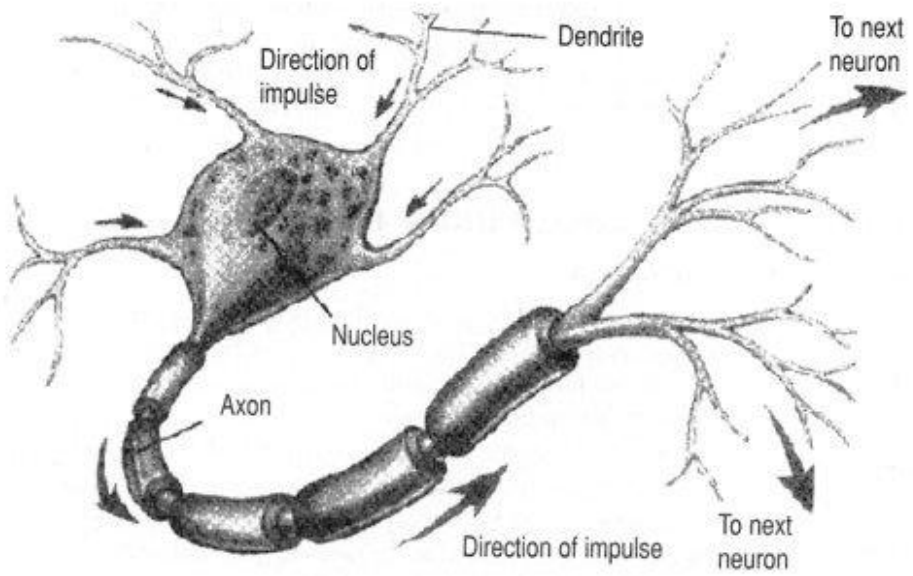
Saagie

Le Deep Learning

Saagie

Neurone biologique

Figure 1: Basic illustration of a neuron



Source: Brown & Benchmark
Introductory Psychology
Electronic Image Bank (1995). Times Mirror Higher Education Group, Inc.

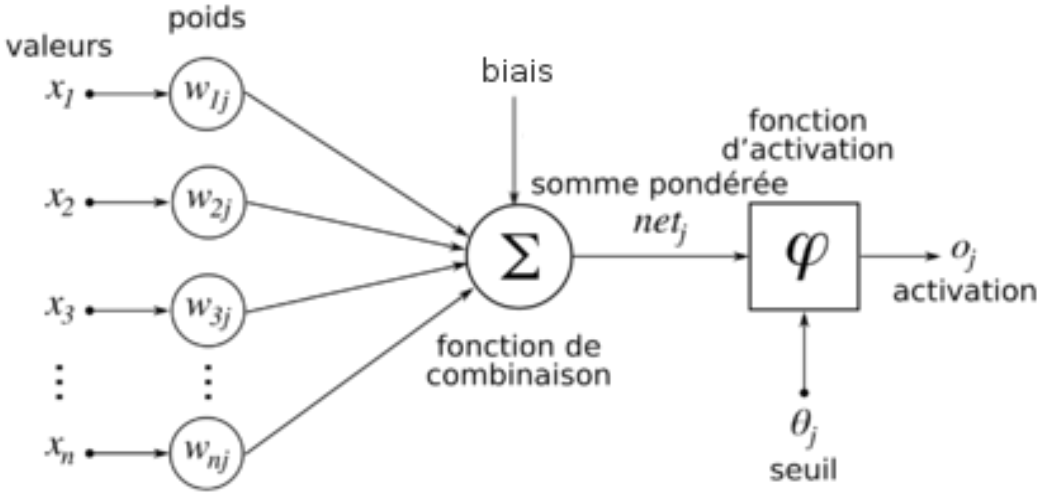
Les dendrites reçoivent un **signal électrique**

Les **Synapses** transforment ce signal

Au delà d'un certain seuil, le neurone s'active (**excitation**) -> **envoi d'un signal** à d'autres neurones

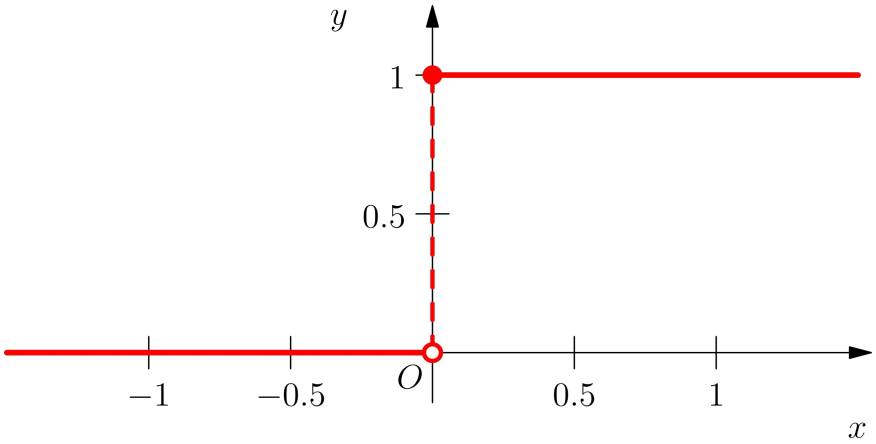
Saagie

Neurone formel (1943)



Représentation **mathématique** d'un neurone biologique

Activation :
Fonction **Seuil**



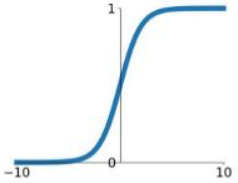
Saagie

Fonctions d'activations

Activation Functions

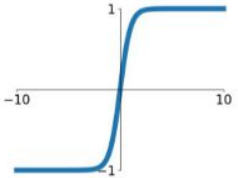
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



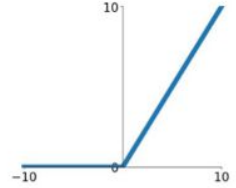
tanh

$$\tanh(x)$$



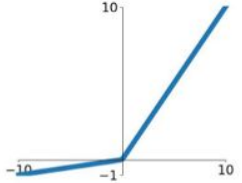
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

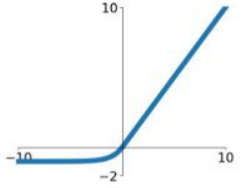


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

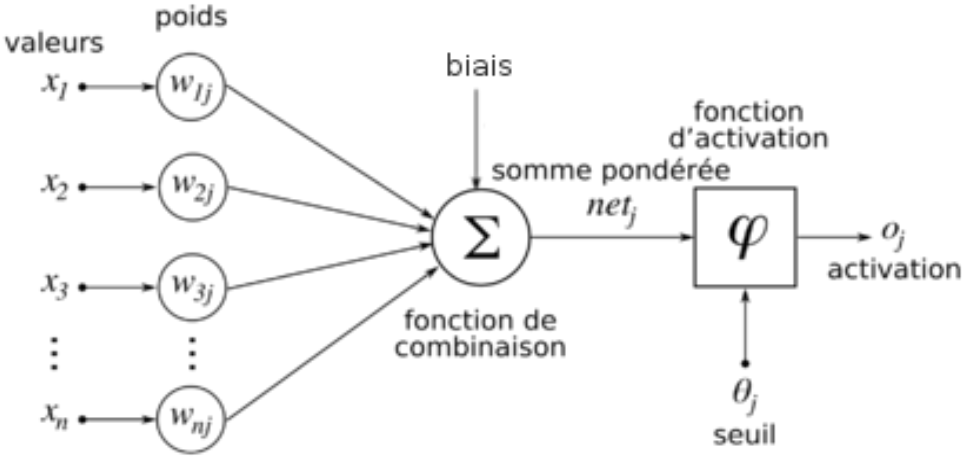
ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Saagie

Perceptron (1957)



Activation = classe 1
Pas d'activation = classe 2

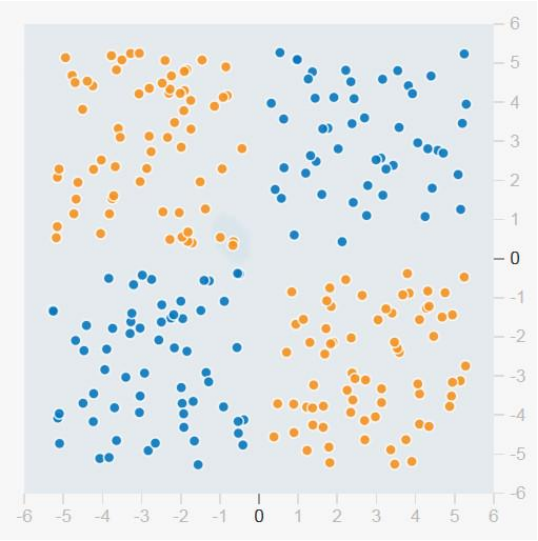
0 ou 1

Perceptron : **Neurone artificiel** utilisé pour faire de la **classification**

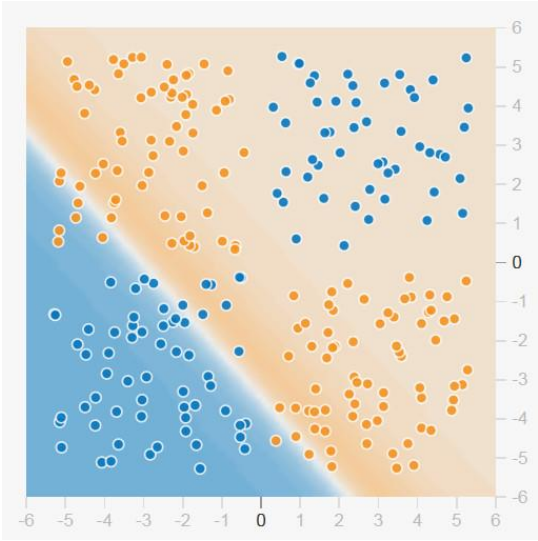
-> **Apprentissage** : Trouver les poids qui marchent

Saagie

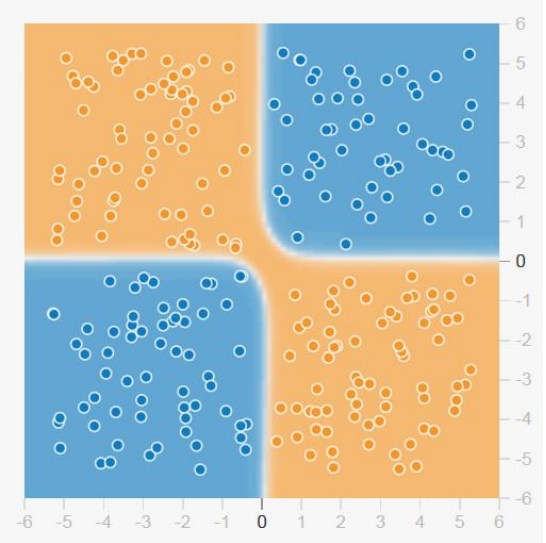
Perceptron simple



Problème XOR



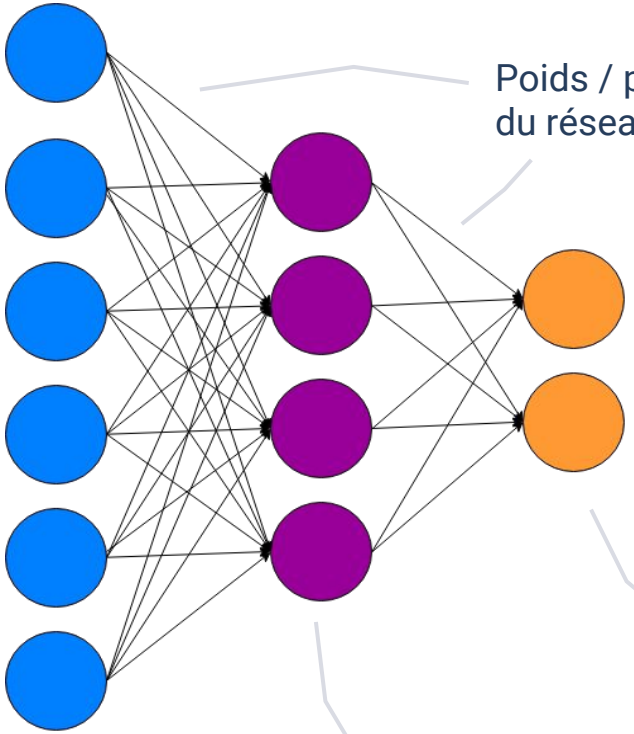
Solution d'un perceptron simple



Bonne solution

Saagie

Réseaux de neurone : Perceptron à une couche



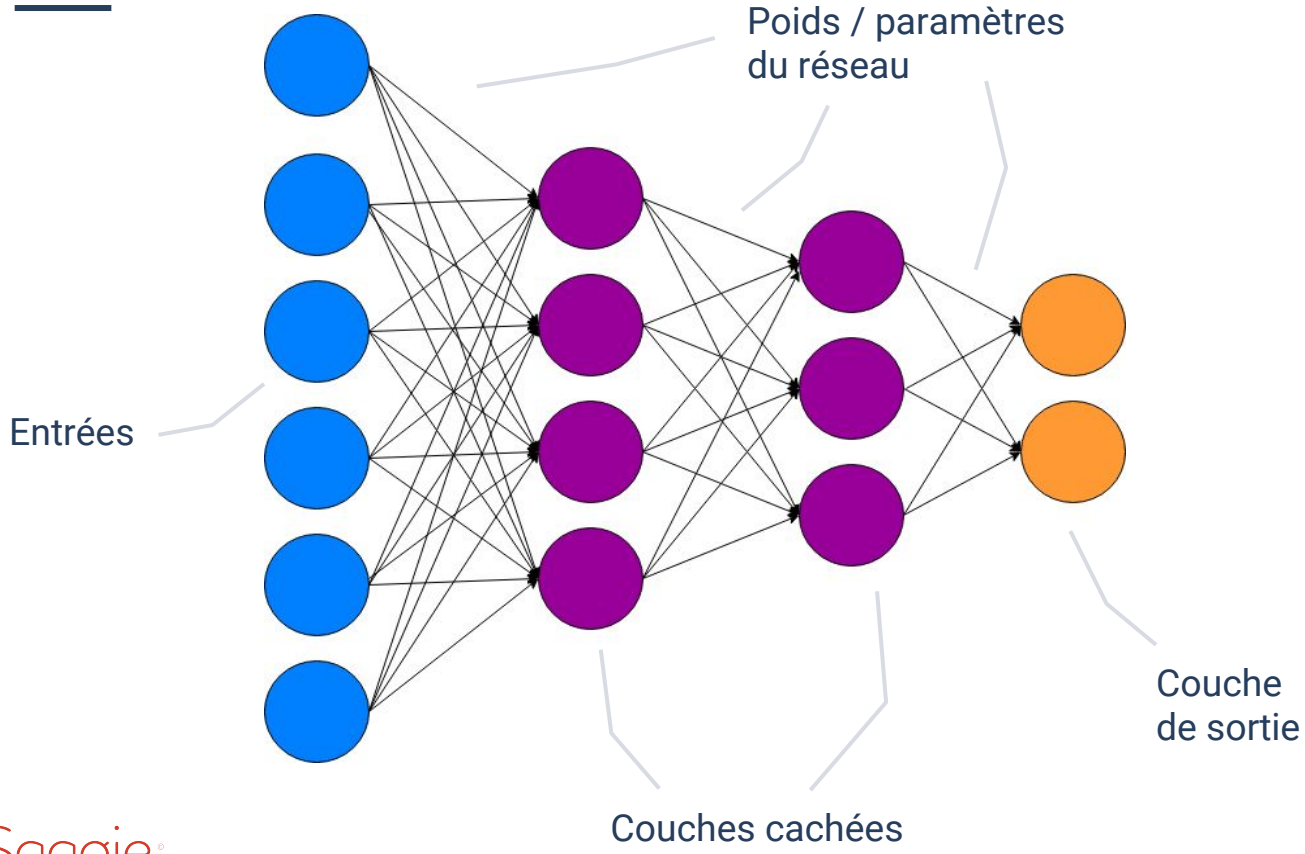
+ Une couche caché

+ Chaque neurone de la couche cachée est connecté à toutes **les entrées**

+ Peut demander **enormement de neurones** sur la couche cachée

Saagie

Perceptron multicouche



+ Besoin de **moins de neurones** au global s'il y a plusieurs couches

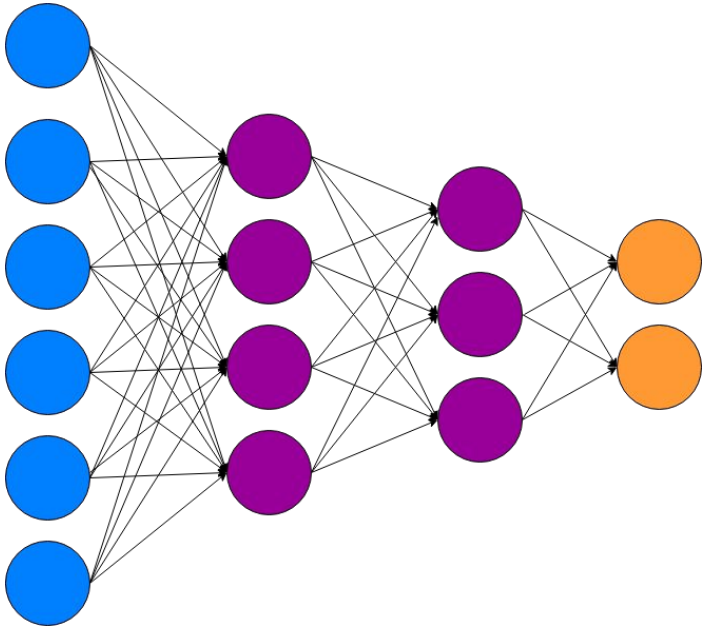
+ Information mieux representée
-> **Hierarchie**

Saagie

Prédiction d'un réseaux de neurones

Transformation des données

Données en entrée



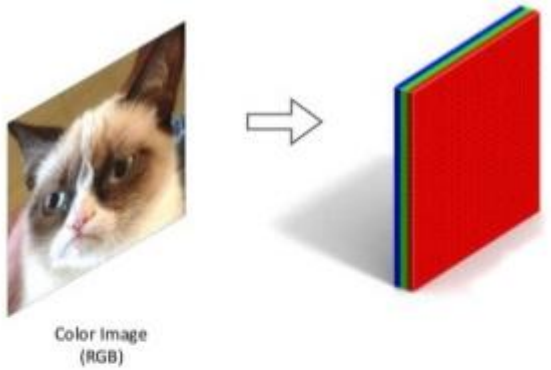
muffin
chihuahua
muffin
chihuahua

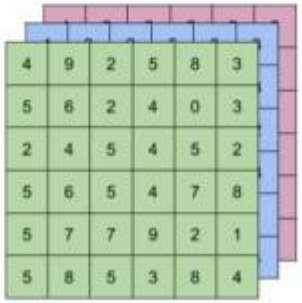
Prédiction

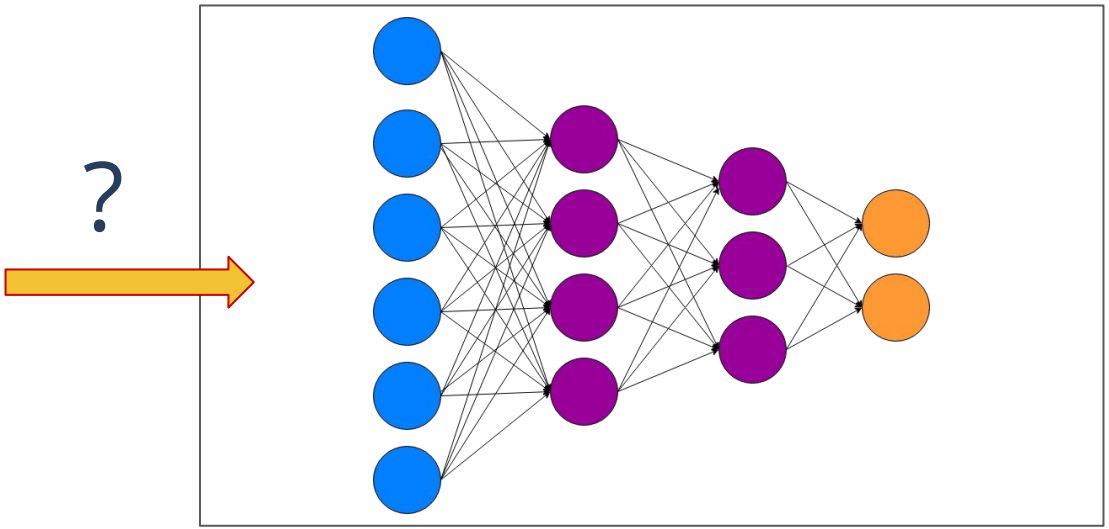
On envoit les images en groupes :
-> **Batch** d'images

Saagie

Deep Learning : Images



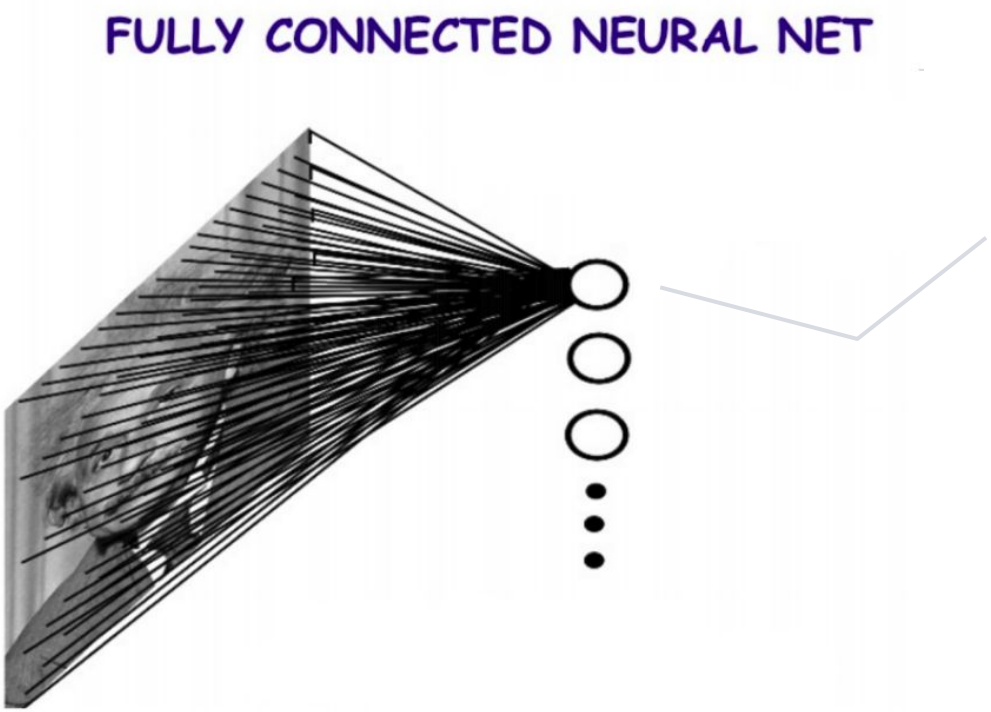




Comment donner une image à un réseau de neurones ?
Image = Tenseur en 3 dimension

Saagie

Deep Learning : Images



Chaque neurone de la 1ere couche est connecté à **chaque pixel** de l'image.

Images 256*256*3
-> 196608 **pixels** en entrée

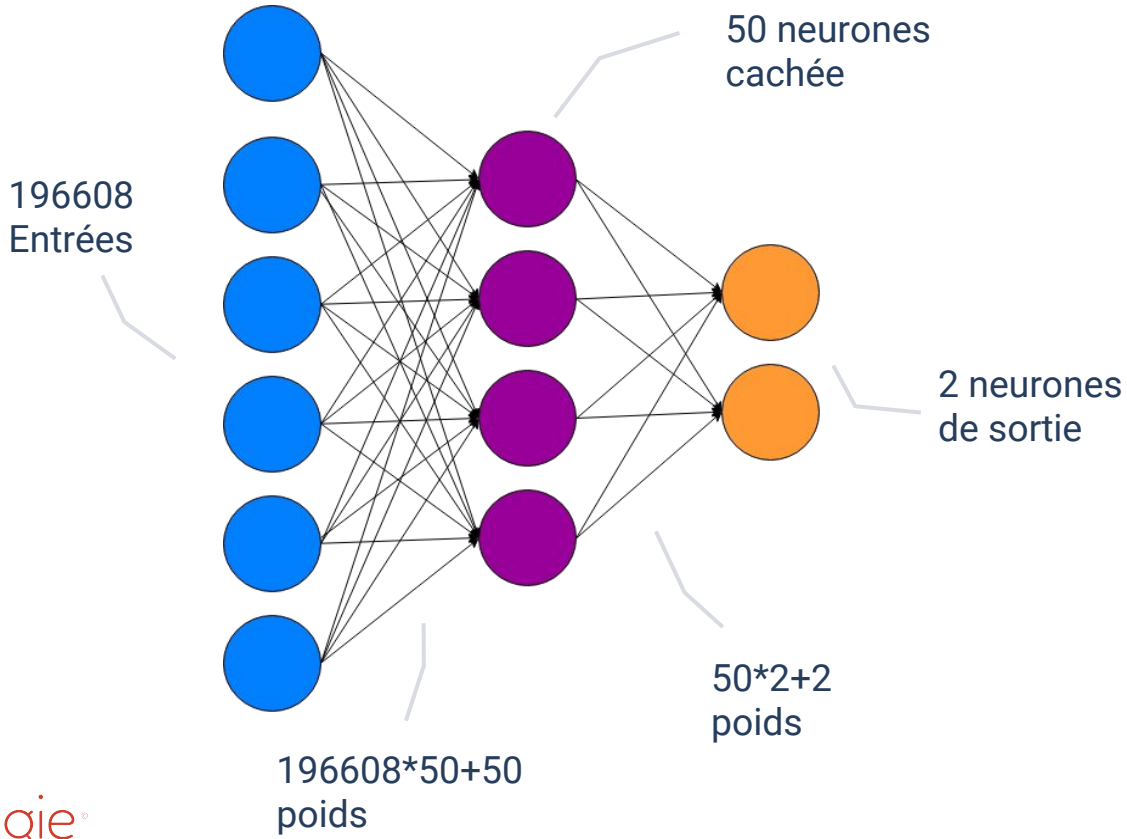
-> Transformation de l'image pour applatir les **3 channels** de couleurs

Deep Learning Frameworks

Deep Learning Frameworks

Pytorch

Apprentissage d'un réseau



Presque **dix millions** de **paramètres** (poids) à trouver

-> Comment faire?

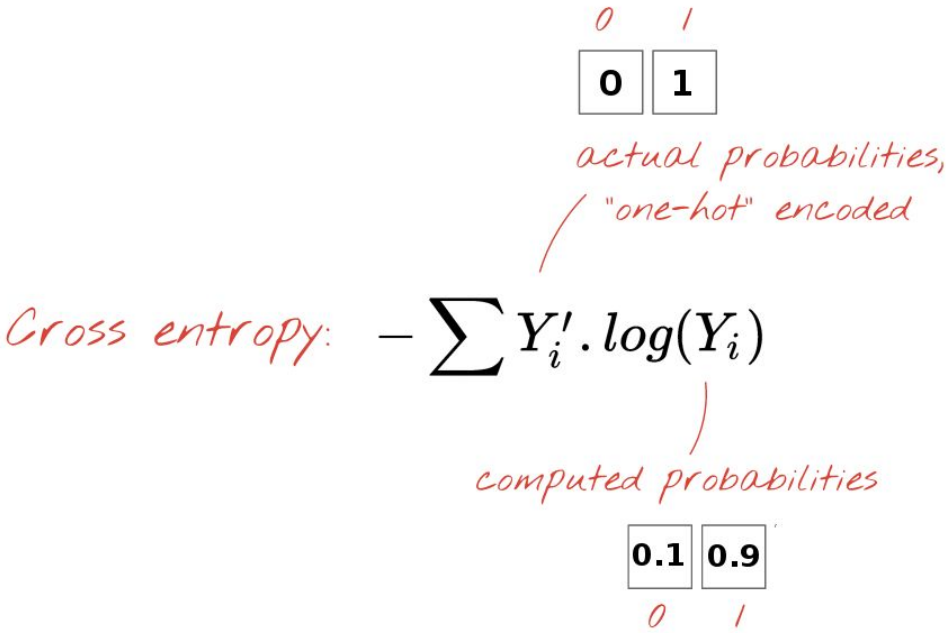
Saagie

Apprentissage d'un réseau de neurones

-> Calculer **l'erreur** :
Fonction cout (loss) : **Distance** entre mes **prédictions** et la **verité** (le label)

-> Objectif : **minimiser** la fonction de coût : réduire les erreurs du réseau

+ Trouver les poids qui rendent l'erreur la plus petite possible
+ Problème : pas de solution analytique
+ Solution : aller petit à petit vers une meilleure solution
  -> Gradient : direction de la solution

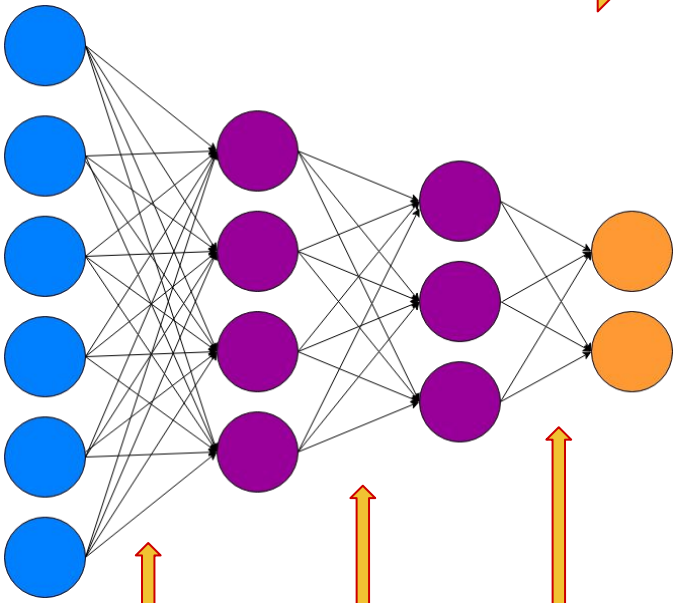


Retropropagation du gradient

Batch de données en entrée

[1 0]
[0 1]
[1 0]
[0 1]
[1 0]
[0 1]

Transformation des données



Propagation des erreur
Mise à jour des poids /paramètres

Prédictions

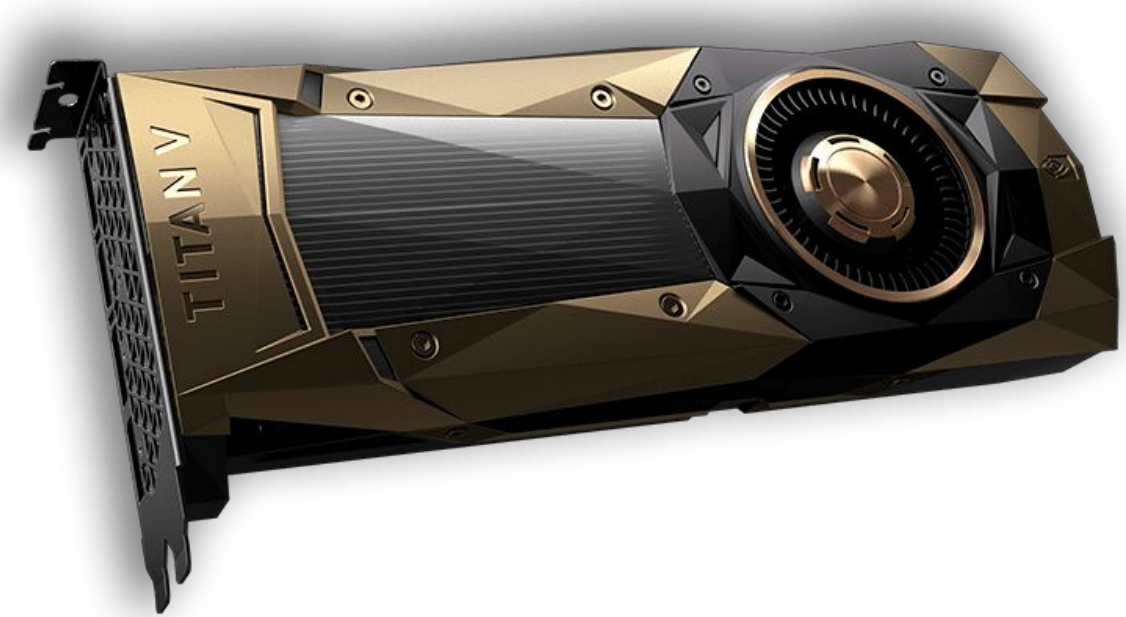
Chihuahua	Muffin
0.7	0.3
0.1	0.9
0.8	0.2
0.4	0.6
0.6	0.4
0.5	0.5

Calcul des erreurs

\+ Une MAJ des poids par batch de données

Saagie

Apprentissage : Puissance de calcul



+ Enormement de petits calculs en parralèle

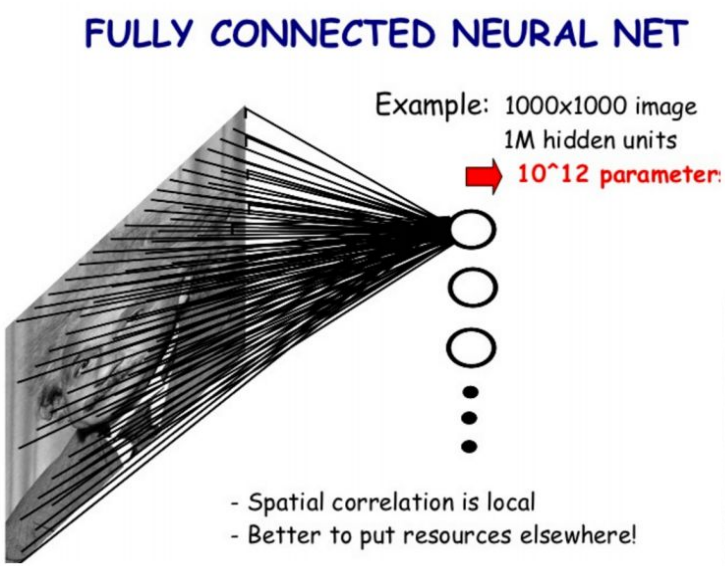
+ Utilisation de carte graphiques : GPUs

+ Apprentissage peut-être très long (jours, semaines, mois) et compliqué

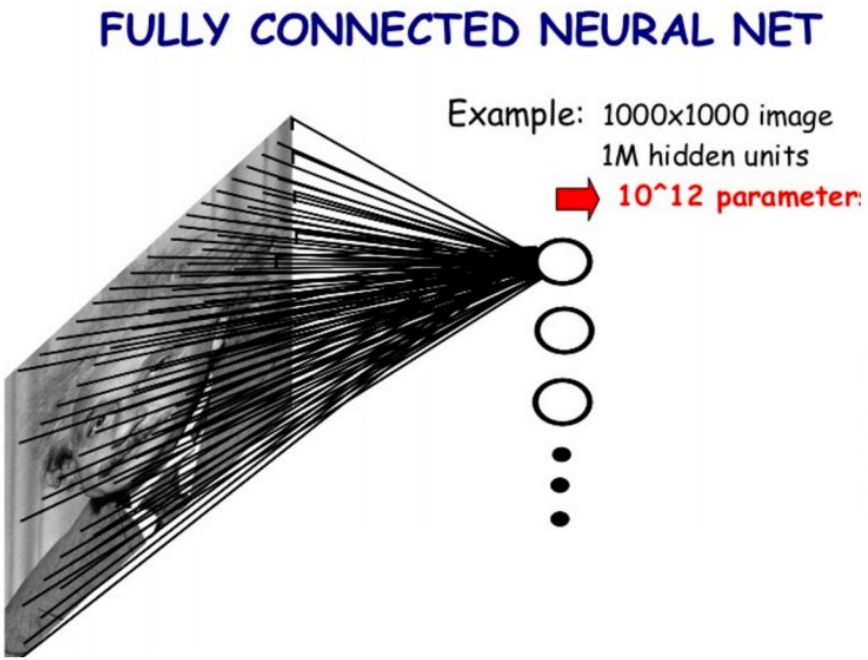
Saagie

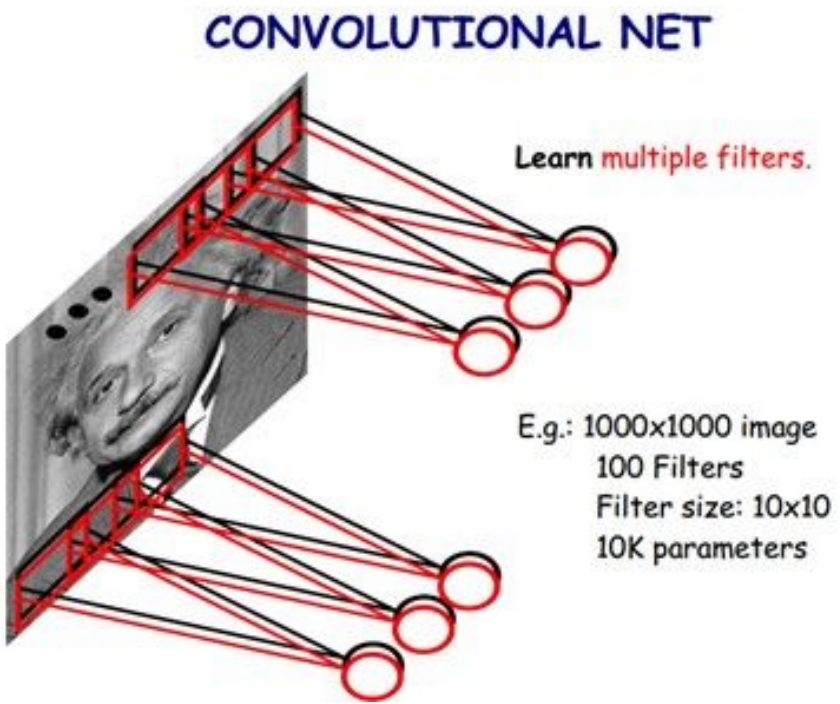
Pytorch

Deep Learning : Images



Fully-connected versus Convolutions on images



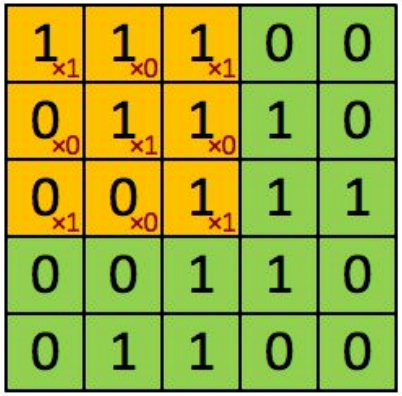


Saagie

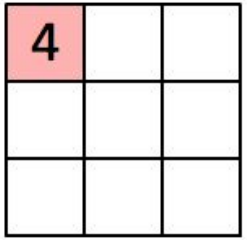
Convolutions

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

Filtre de convolution



Image

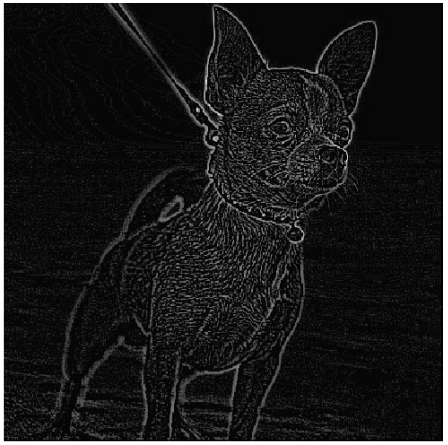


Convolved Feature

Convolutions : filtres

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

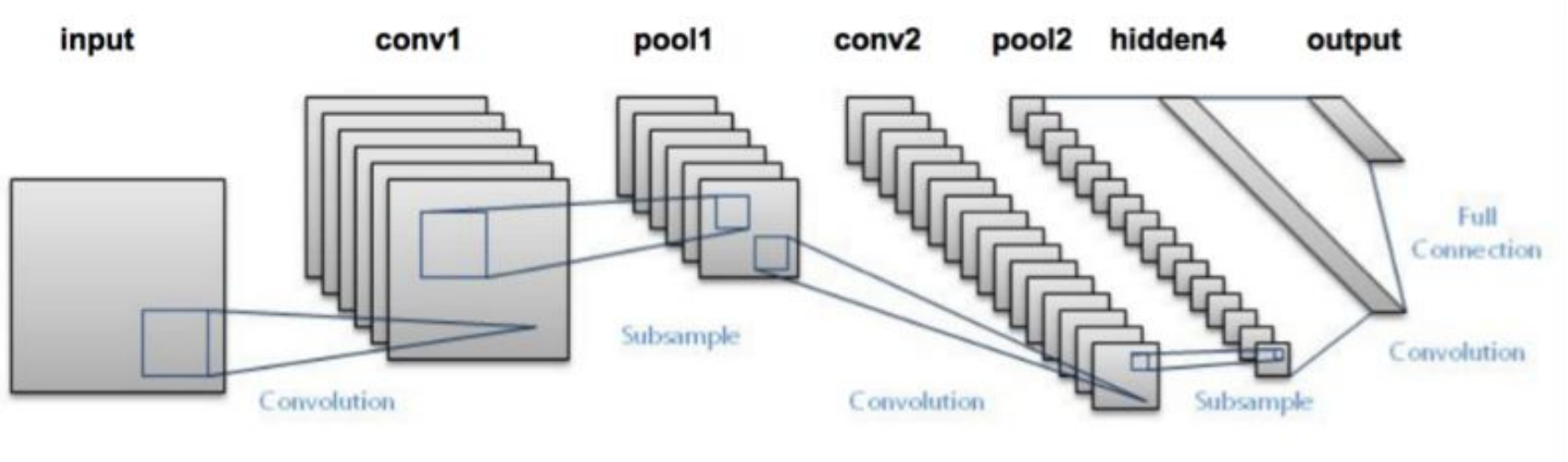
Detecteur de contour



+ Réseau convolutionnel -> Beaucoup de filtres appliqués
+ Apprentissage : Trouver les filtres adaptés pour résoudre le problème

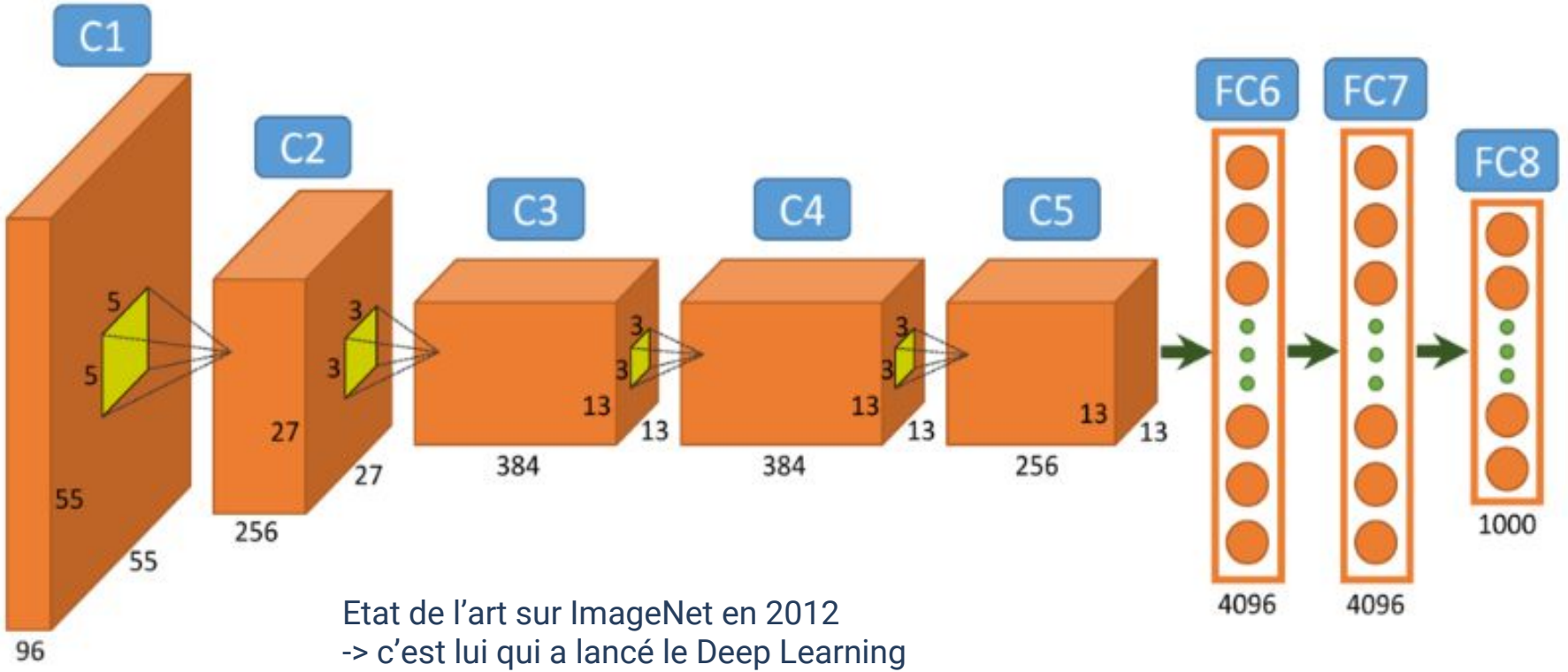
+ Hiérarchie de représentation des filtres

Réseau convolutionnel : LeNet



+ Developpé fin 80 / début 90
+ Reconnaissance de caractères sur les cheques
+ Architecture de base des réseaux de neurones convolutionels

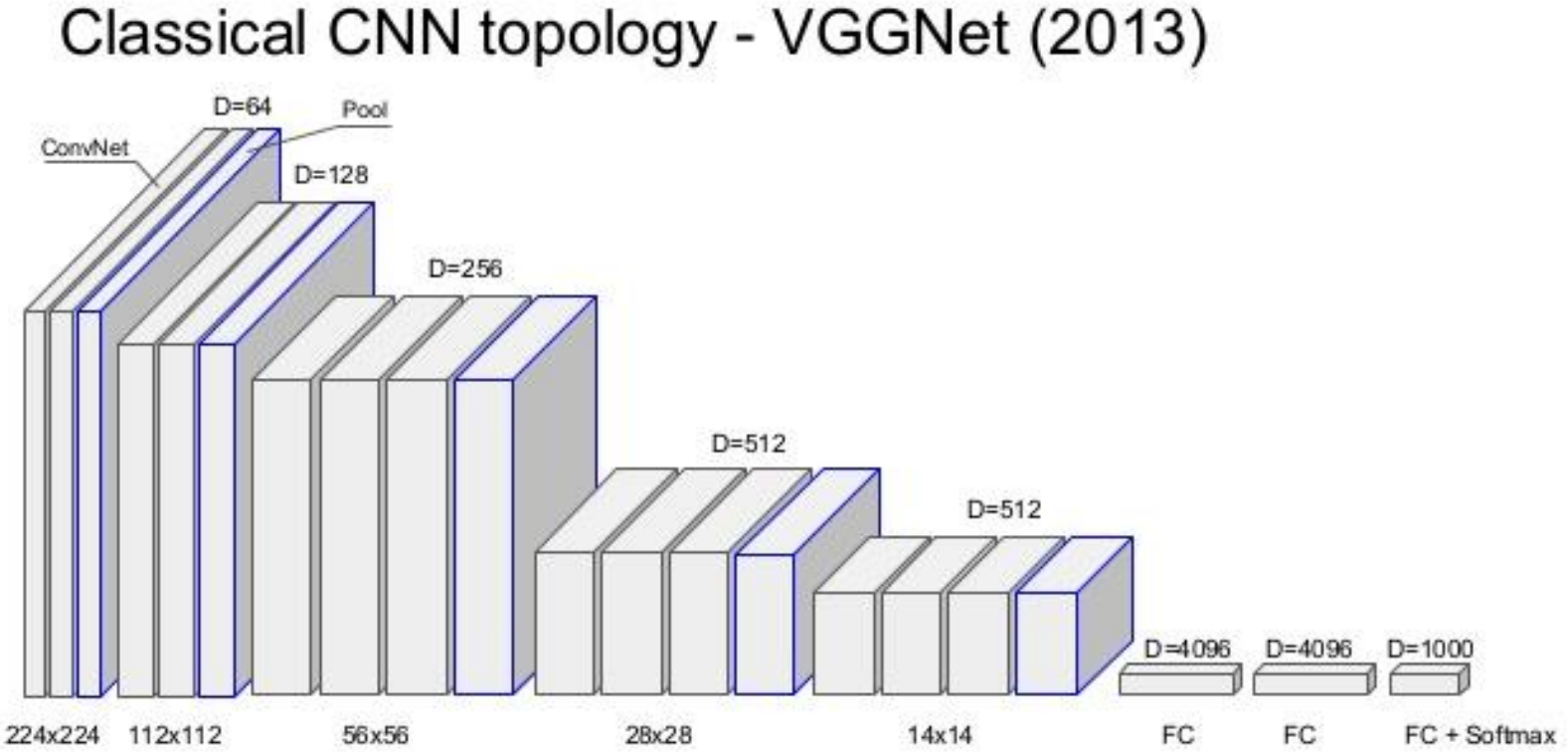
A popular CNN : Alexnet 2012



Etat de l'art sur ImageNet en 2012
-> c'est lui qui a lancé le Deep Learning
-> 60 millions de paramètre

Saagie

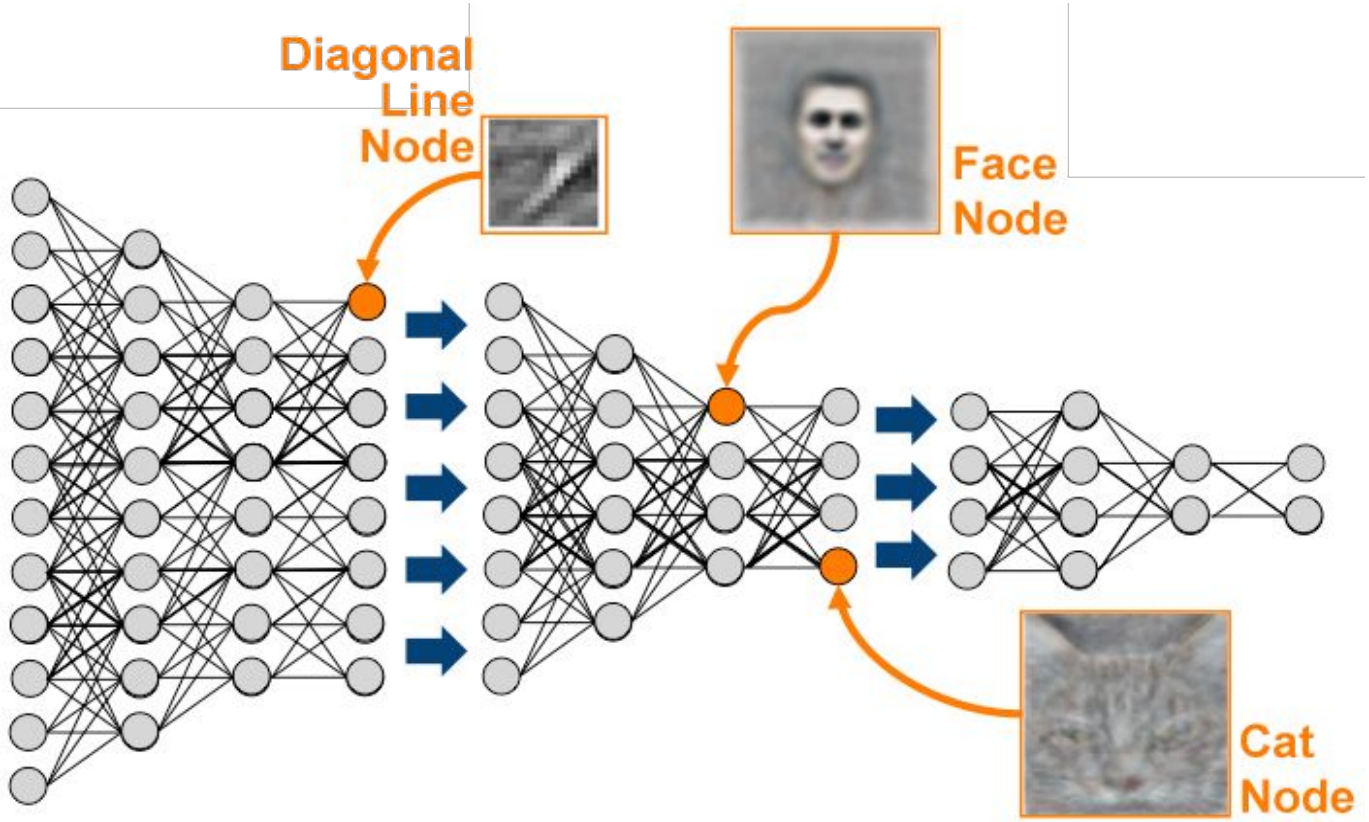
VGGNet



Saagie

Pytorch

Apprentissage par représentation

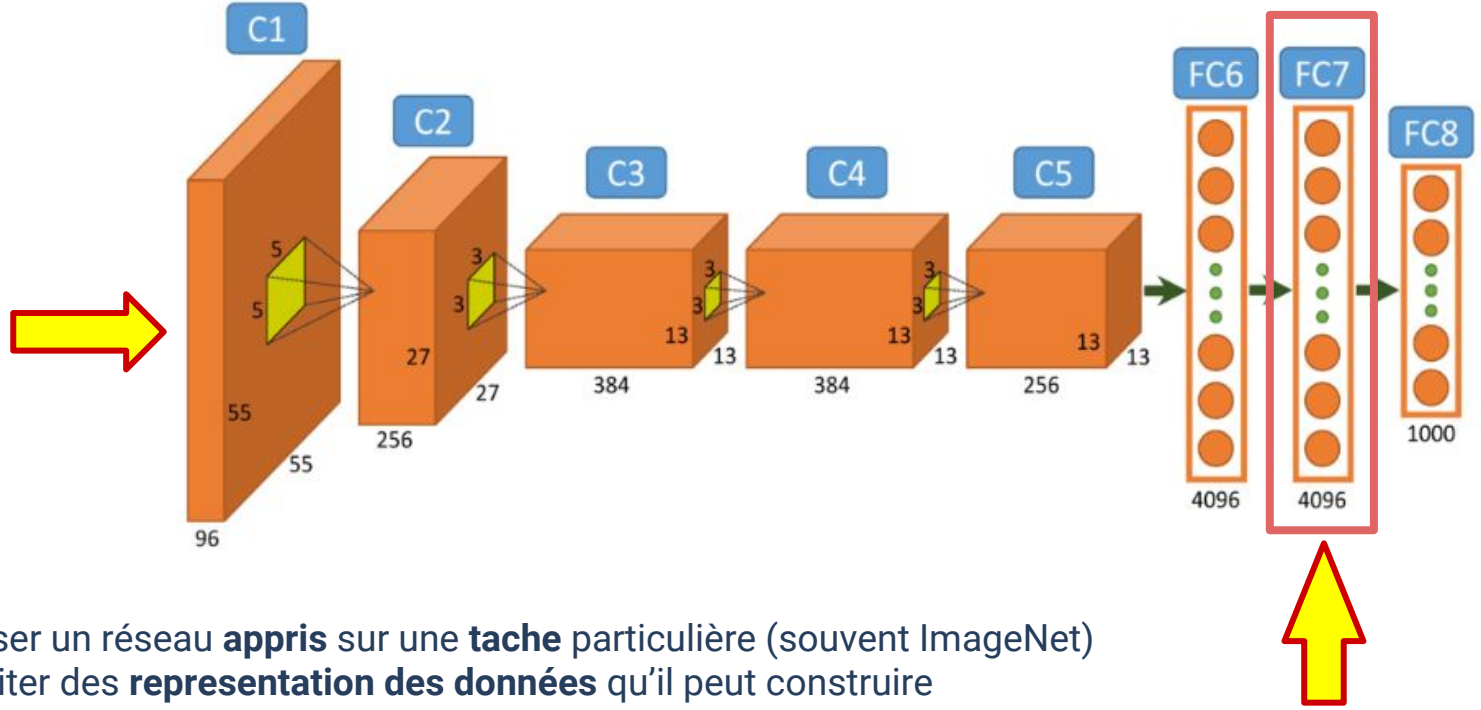


+ Les neurones se specialisent

+ Des representations sont apprises

Saagie

Representation Learning

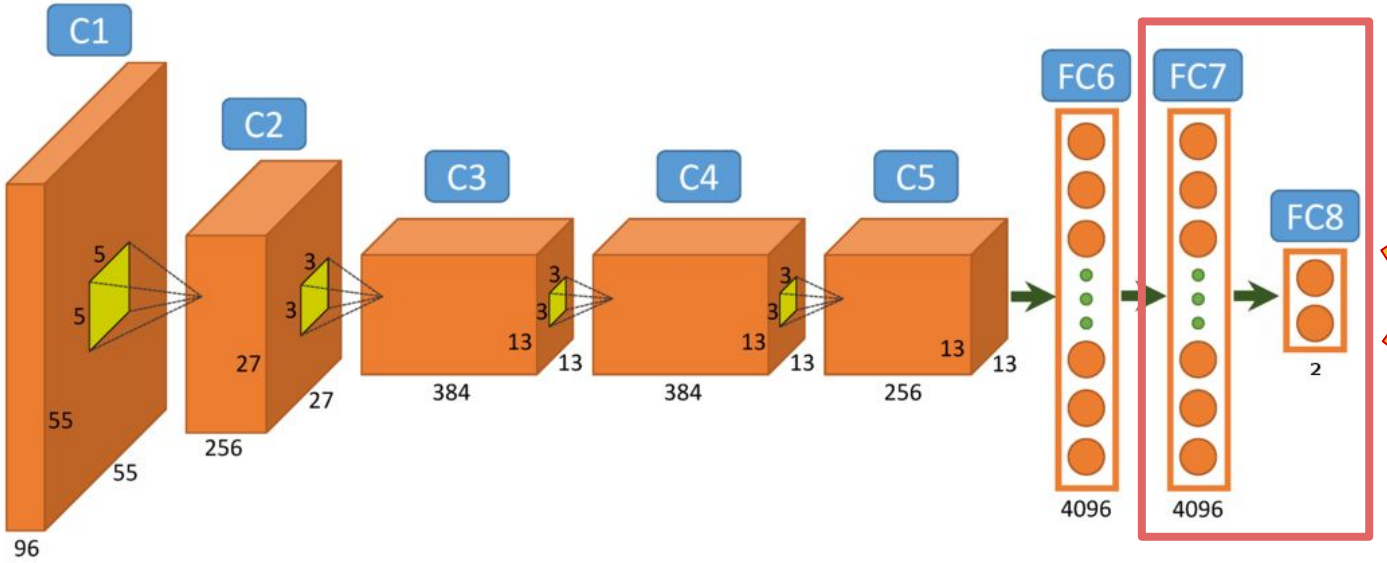


+ Utiliser un réseau **appris** sur une **tache** particulière (souvent ImageNet)
+ Profiter des **representation des données** qu'il peut construire

Features extraction

Saagie

Transfert Learning - Fine Tuning



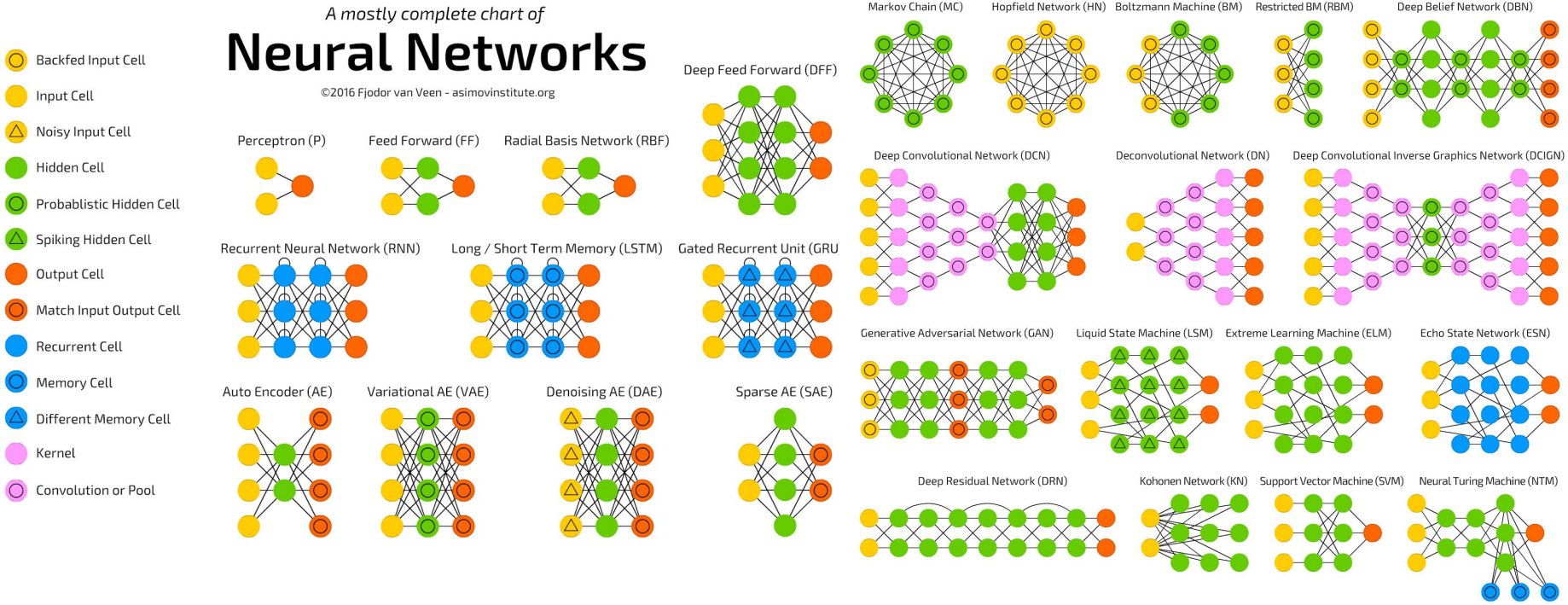
+ Prendre un réseau **pré-appris**
+ Le “**tuner**” pour lui apprendre une autre tâche (similaire)

Saagie

Autres cas d’usages

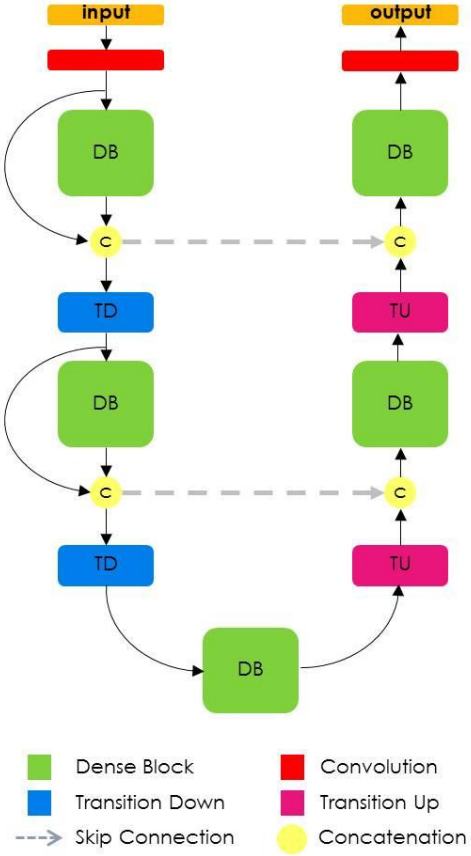
Saagie

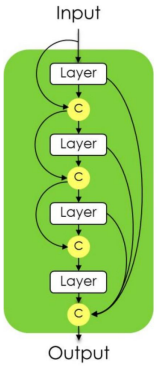
Quelques exemples de réseaux



Saagie

Tiramisu





Saagie

NicolasCaging

Saagie

DeepFakes

Saagie

FaceSwap : Dataset






Saagie

FaceSwap V1

Saagie

FaceSwap V2

Saagie

FaceSwap V3

Saagie

Merci

CONTACT +(33)2 72 88 31 69 | contact@saagie.com | www.saagie.com

Saagie©

Seine Innopolis
72, rue de la République
76140 Le Petit-Quevilly

BPI Paris
8 Boulevard Haussmann
75009 Paris

Galvanize San Francisco
44 Tehama St
San Francisco CA 94105