L’intelligence artificielle

Les réseaux de neurones

« deep learning »

Intelligence artificielle :

Selon les fondateurs de la cybernétique, le terme « intelligence artificielle » désigne un comportement produit par une machine dont on peut raisonnablement estimer que, s’il avait été le fruit d’une action humaine, il aurait exigé de l’intelligence de la part de l’agent concerné.

Source : wikipedia

Remarque : une calculatrice est une IA

Yann Le Cun

Titulaire pour l'année de la chaire « Informatique
et sciences numériques » du Collège de France.
Scientifique en chef de l'IA chez Facebook.

Stephane Mallat

Professeur du Collège de France sur la Chaire de Sciences
des Données.

Hugo Larochelle (video you tube)

Un peu d’histoire…

Alan Turing (1912-1954) : concevoir une machine comme un enfant (test de Turing)

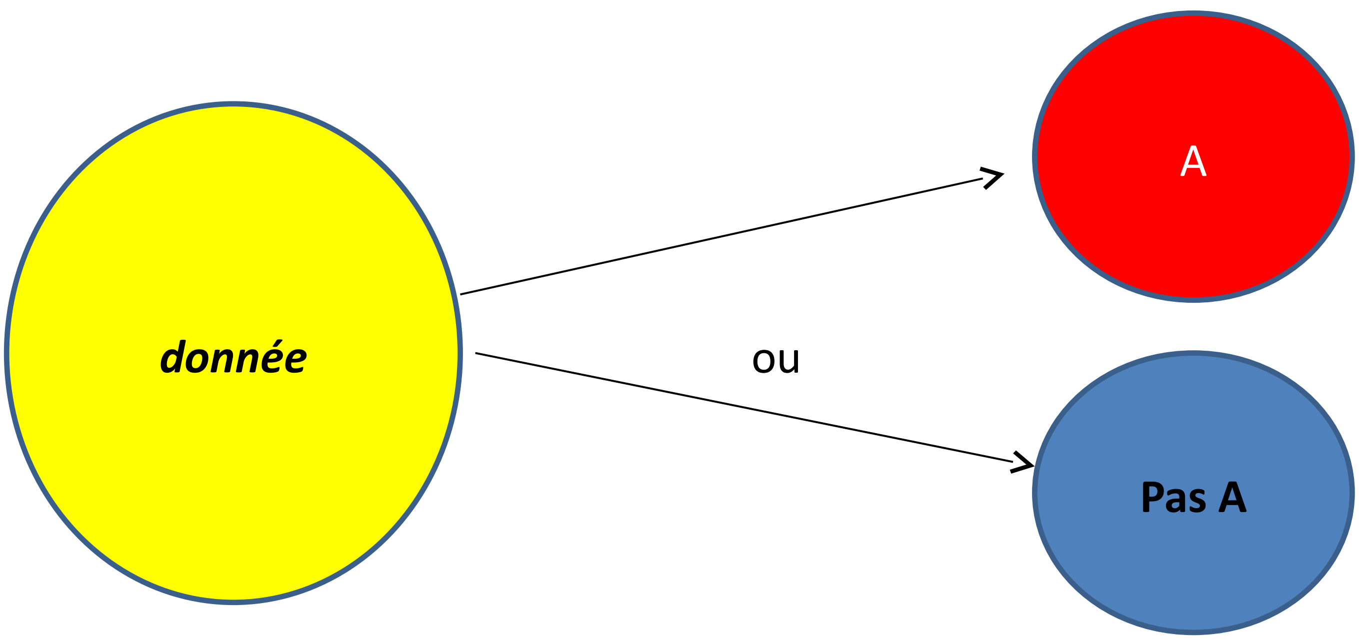
1956 : machine neuronal, apprentissage rudimentaire
1980 : retropropagation du gradient (victoire aux échecs)
2012 : Avènement de l’IA : réseaux convolutifs, (bcp données, machine puissante)

A quoi ça sert (actuellement et encore...)

- Vision par ordinateur (Imagerie médicale – reconnaissance faciale ou de caractère – détection de comportement -aide a la navigation)
- Traitement du langage et de la parole (Traduction- chatsbots – traitement de la voix, aide aux personnes mal entendante...)
- Robotiques https://www.youtube.com/watch?v=sronAgjRHJw
- Physique (calcul d'énergie, problème a N-corps, ...)
- Prédiction de comportement (élection, commerce, recrutement...)
- Jouer a des jeux (échec, go, …)
- Militaires
- …

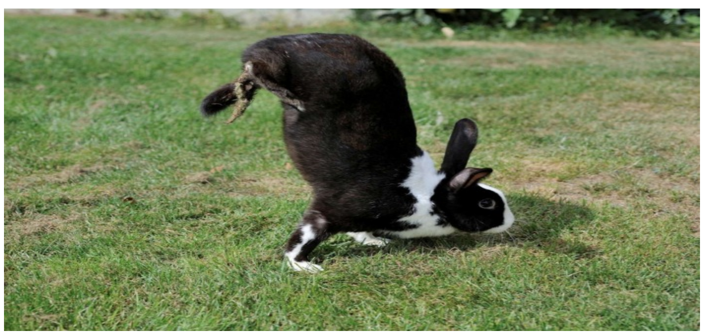
Prise de décision et prévision

Catégoriser des données

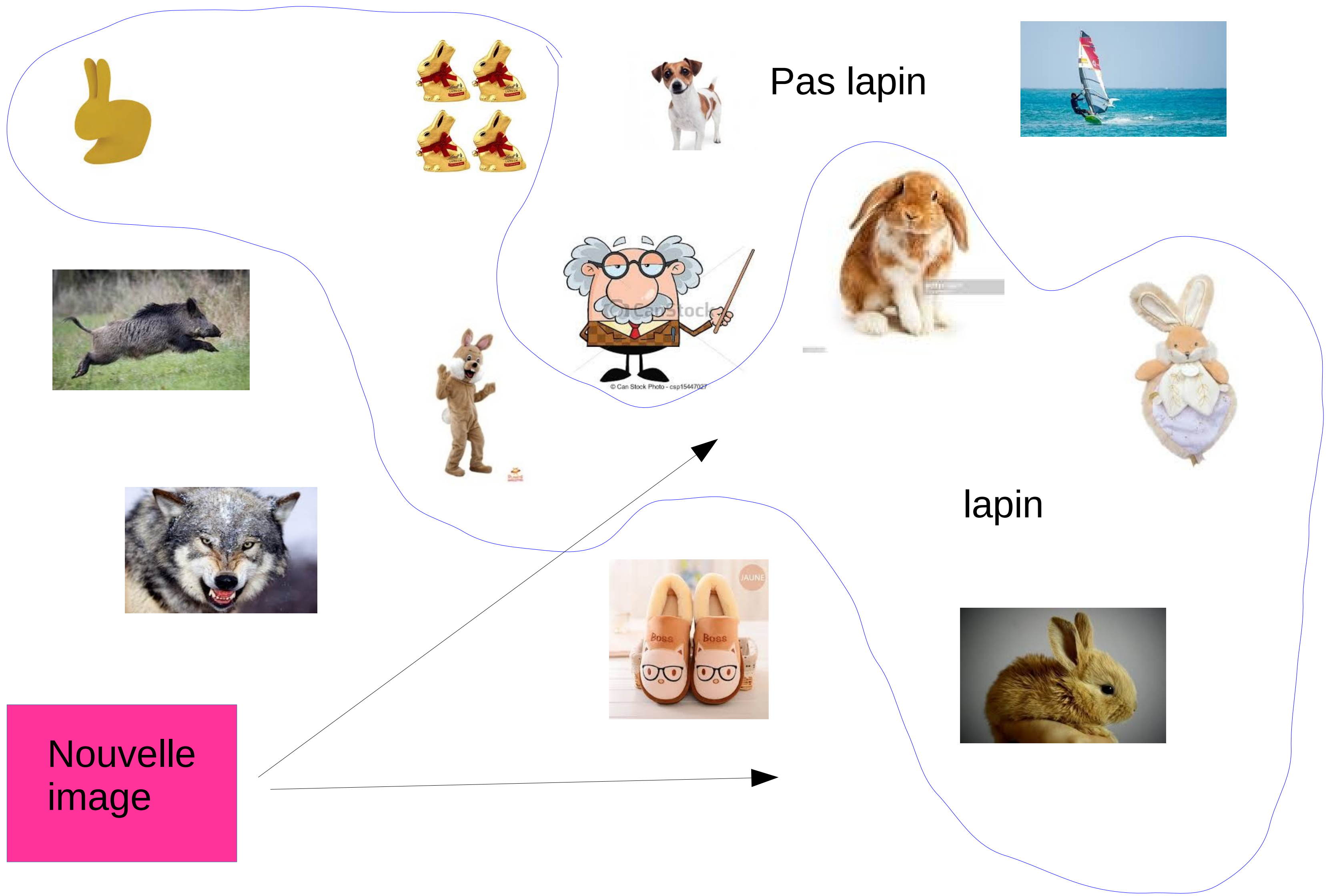


IMAGERIE MEDICALE

Probleme : reconnaître les photos de lapins

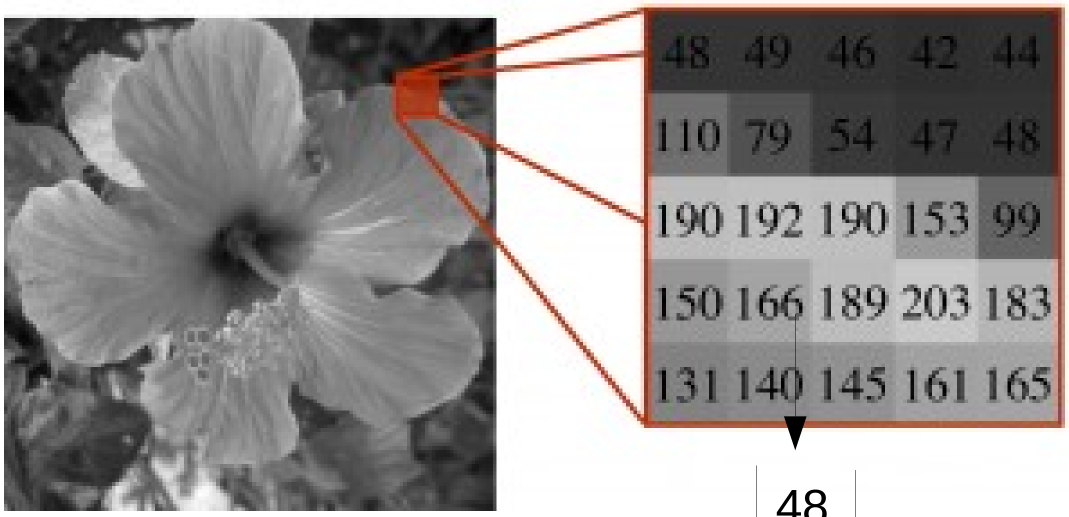


On a des données (par exemple fournie par internet)



Comment entrer une photo dans un ordinateur ?

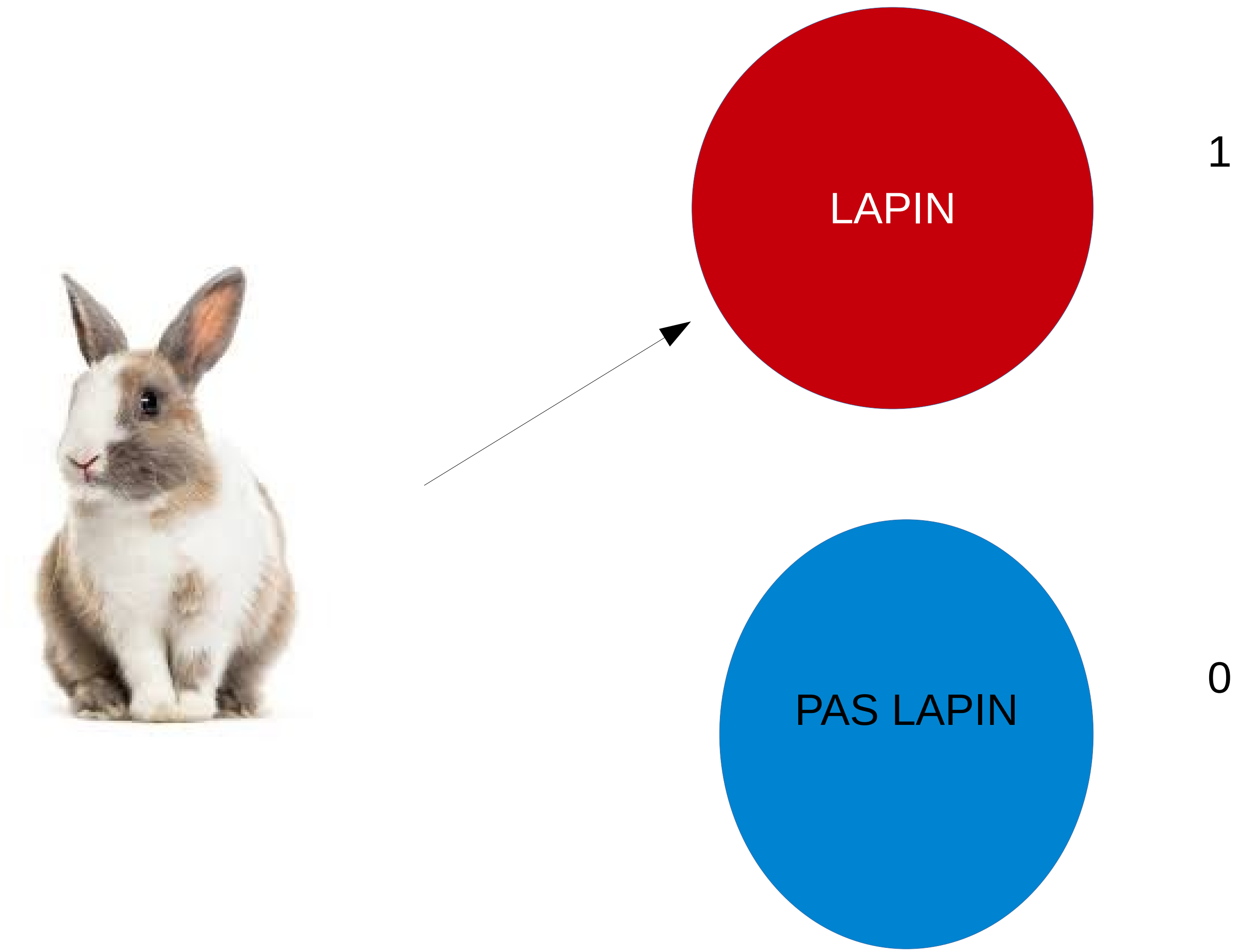
Une photo est un VECTEUR (une colonne de nombre) dans un Espace a n dimensions (n est la taille de la colonne)

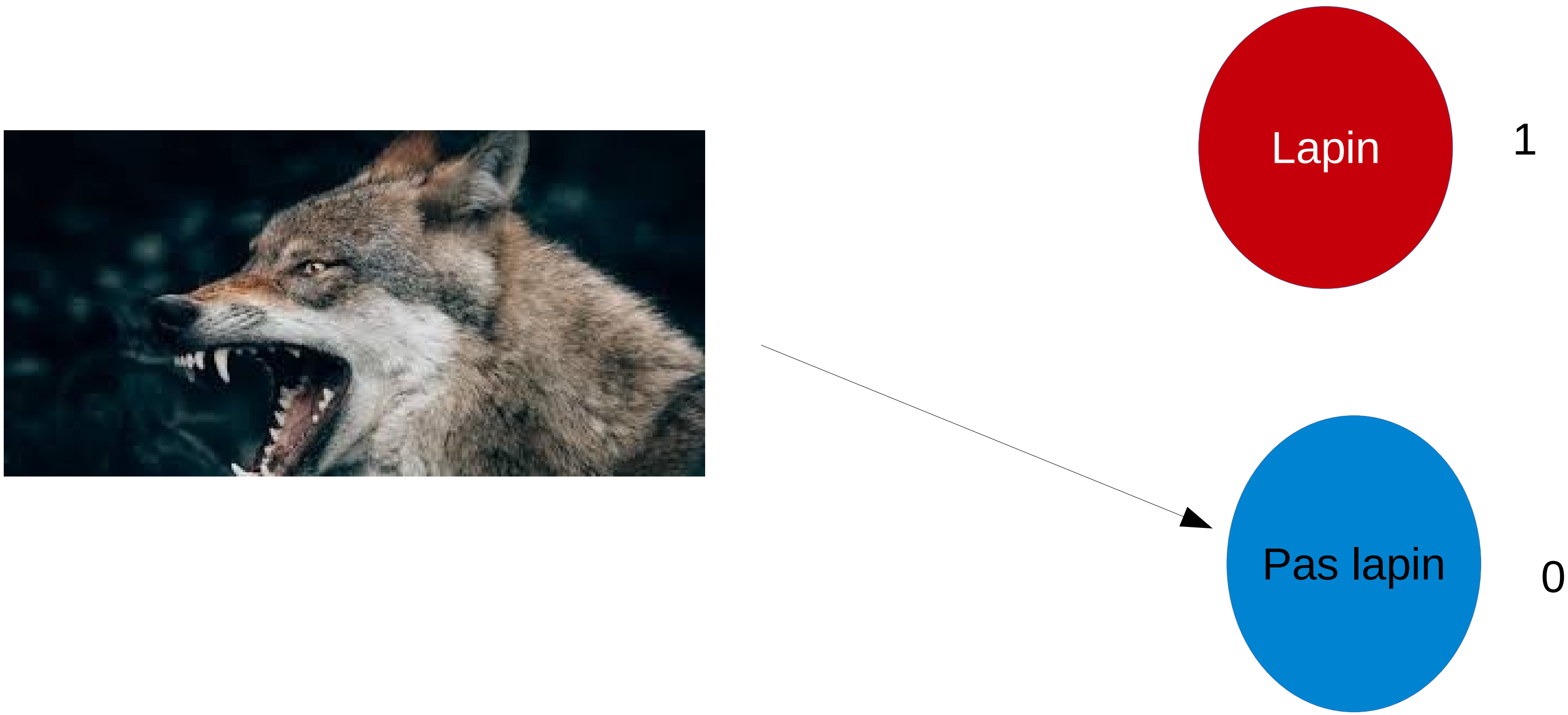


48
49
46
42
44
110
79
54

Photo de 30 sur 30 pixels
N=900 !!

Une sortie est aussi un vecteur




Lapin
1
Pas lapin
0

Reconnaître des lapins c'est trouver une fonction F telle que :

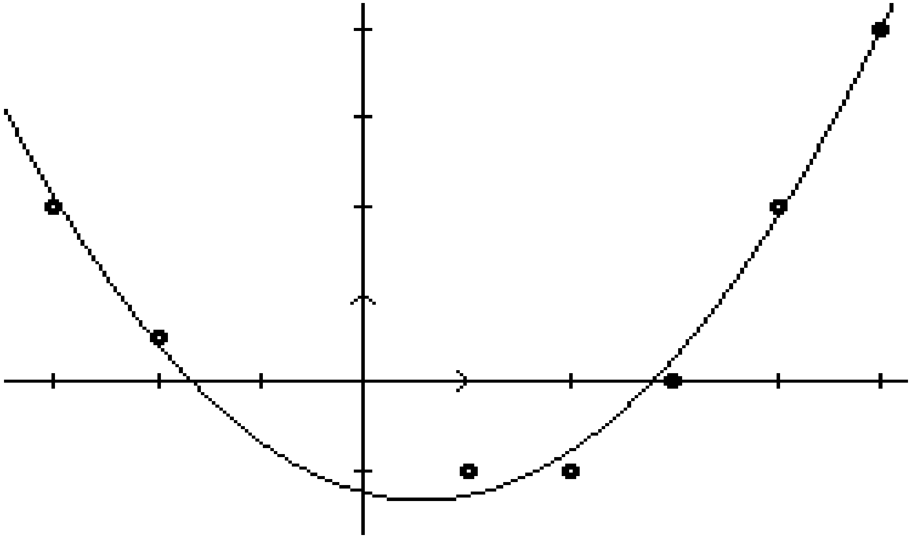
$$F(v) = \begin{cases} 1 & \text{si } v = \text{photo de lapin} \\ 0 & \text{si } v = \text{photo de pas lapin} \end{cases}$$

Problème : Comment déterminer F

(existe-t-elle, est-elle unique?)

Comment déterminer la fonction F ?

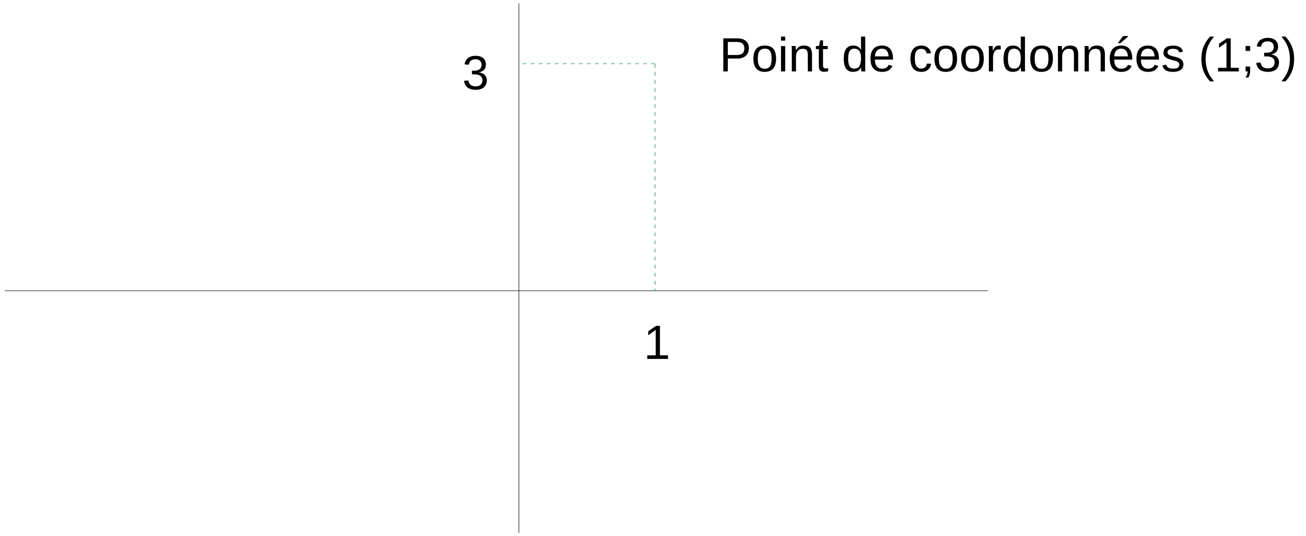
On a des données, c'est à dire des vecteurs v pour lesquels on connaît la sortie : problème d'interpolation

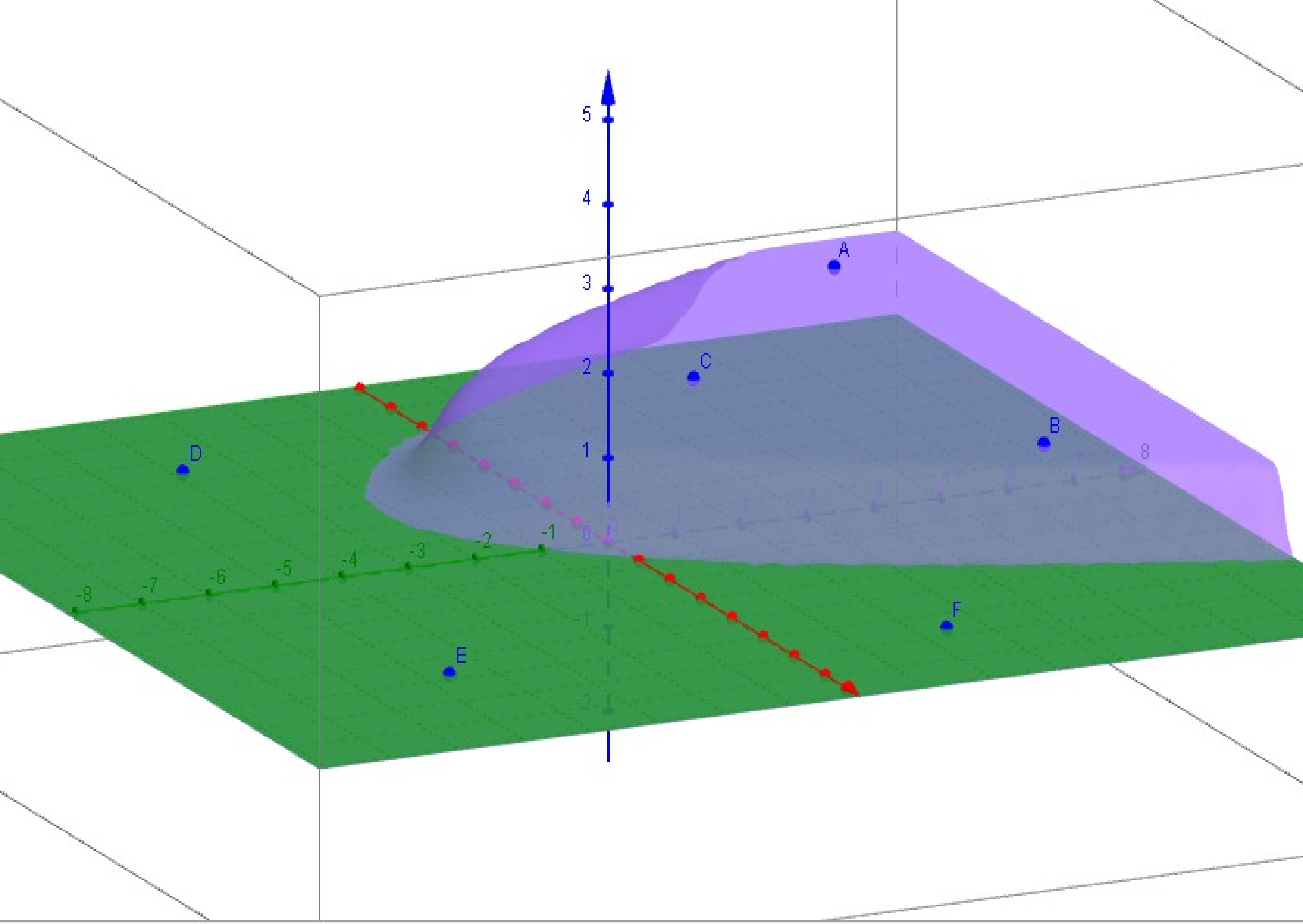


Et on espère que la fonction trouvée sera la bonne !

Remarque : un vecteur de taille n est un point dans un espace de dimension n

Supposons n=2 : une photo est alors un point dans le plan

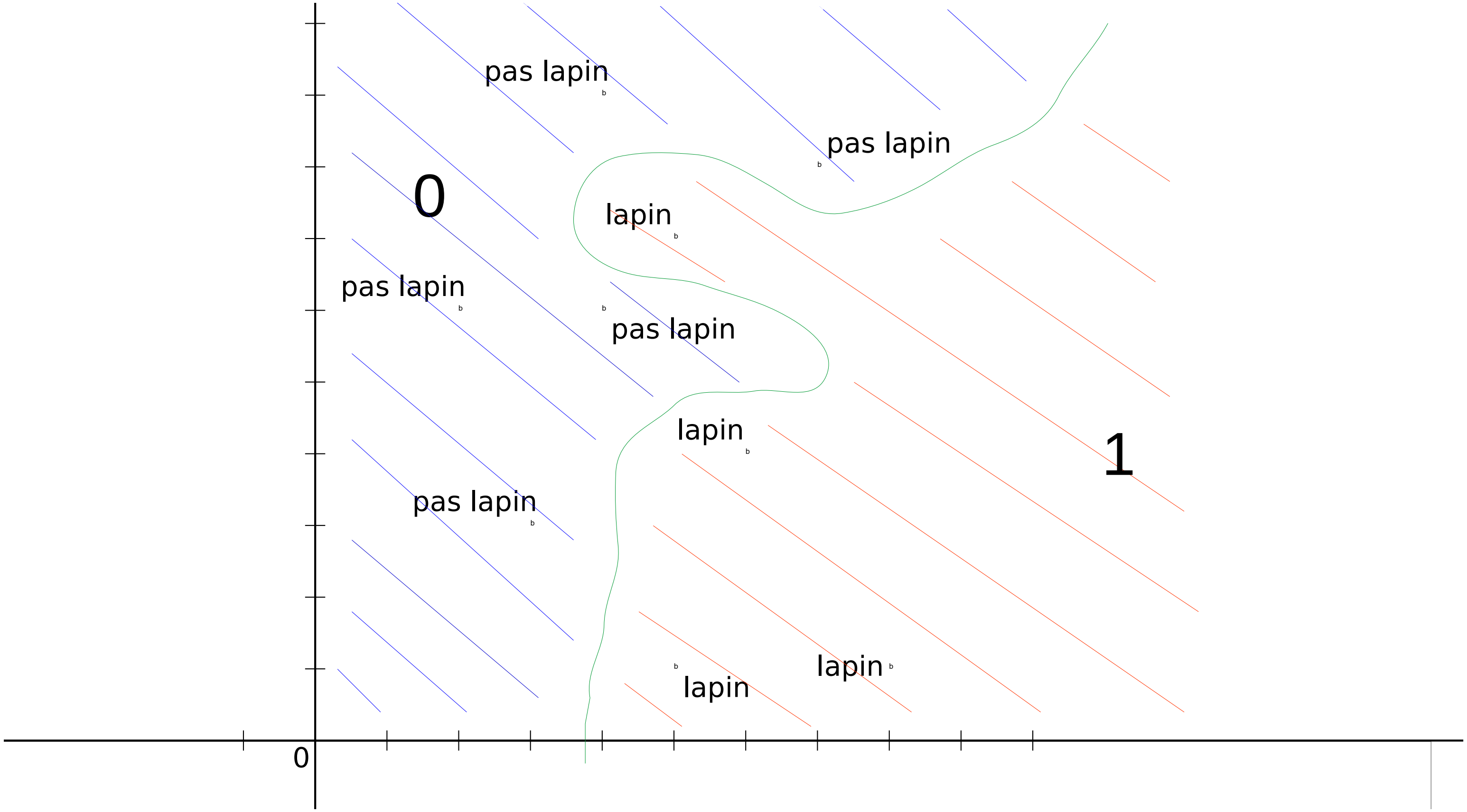




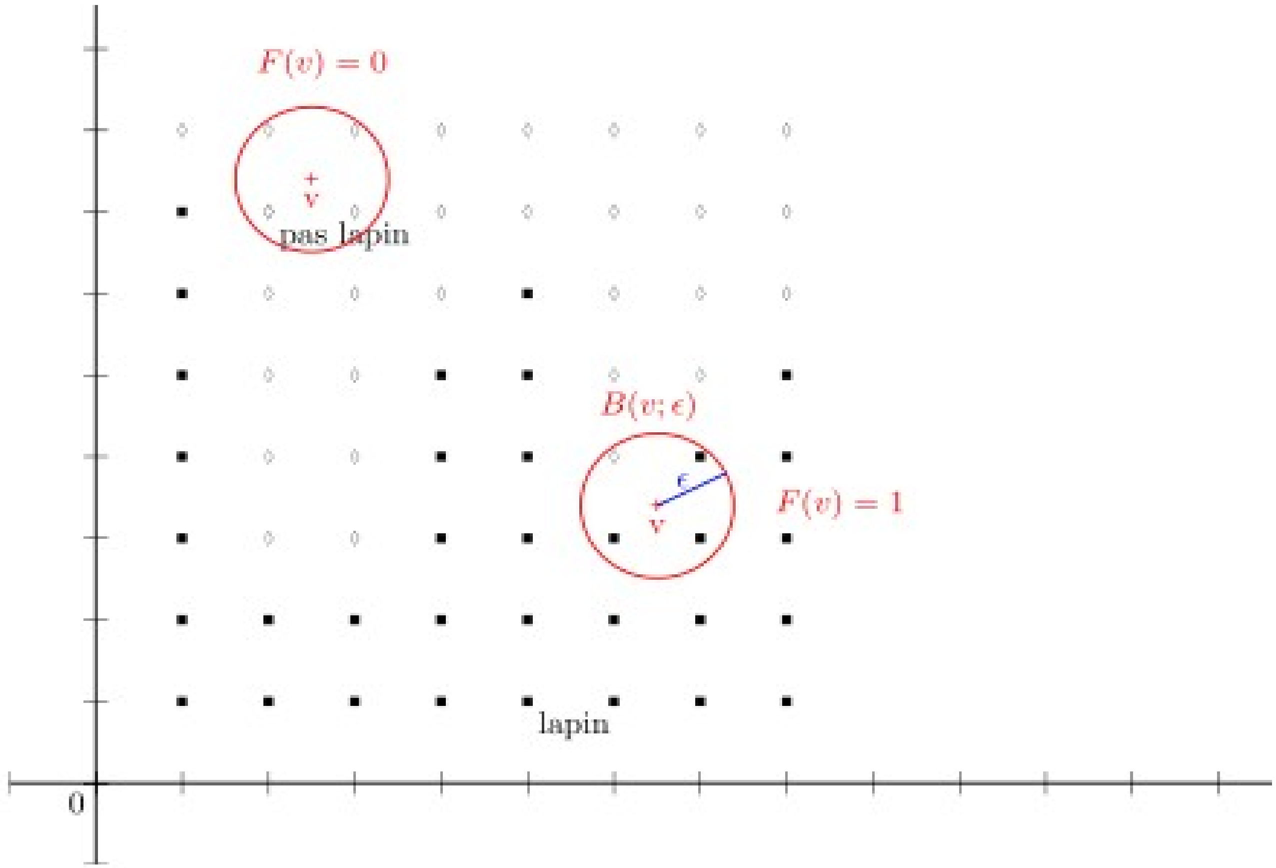
5
4
3
2
1
0
A
B
C
D
E
F
-8
-7
-6
-5
-4
-3
-2
-1
8

Les données sont des points dans un espace de dimension n

$$\varphi : (x;y) \mapsto \begin{cases} 1 & \text{si } (x;y) \text{ lapin} \\ 0 & \text{sinon} \end{cases}$$



Si on a beaucoup d’exemples, c’est un probleme assez simple...



$$F : v \mapsto \begin{cases} 1 \ \text{si dans} \ B(v;\epsilon) \ \#\text{lapin} \geq \#\text{paslapin} \\ 0 \ \text{sinon} \end{cases}$$

Mais a-t-on vraiment beaucoup d’exemples ?

Image de 30 sur 30 pixels : le vecteur de l’image a 900 coordonnées

Si chaque coordonnées prend 10 valeurs alors on a :

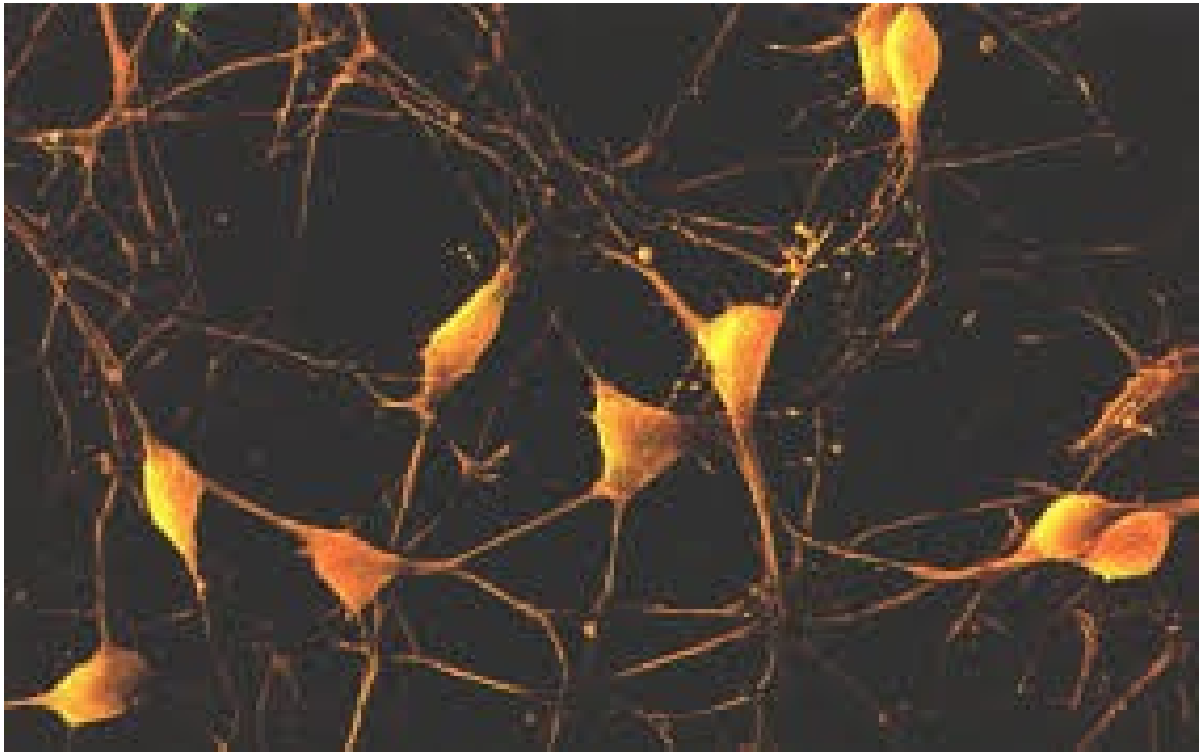
10^900 images possibles

Si 10^9 exemples alors la proportion d’exemples est de :

1/10^881

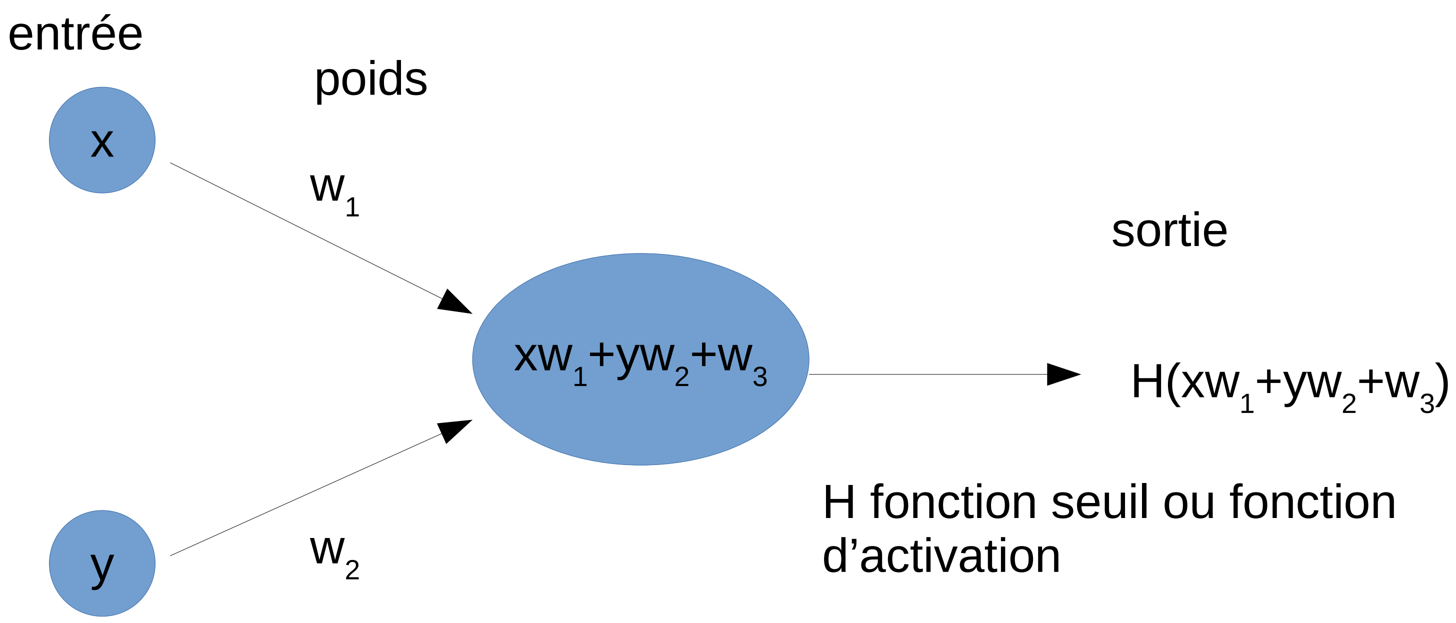
La distance moyenne entre deux images est d’environ de 12 unités

Réseau de neurones



LE PERCEPTRON (ou neurone)

ou comment trouver cette fonction



H fonction seuil ou fonction d'activation

$$H(x)= \left| \begin{array}{ll} 1 & \text{si } x>0 \\ 0 & \text{si } x<0 \end{array} \right.$$

(Remarque : on veut une fonction H dérivable)

Chaque choix des poids donne une nouvelle fonction : il faut donc choisir les poids

Comment déterminer les bons poids ?

On cherche les poids w_1, $w_{2,}$ w_3 tels que

le coût :

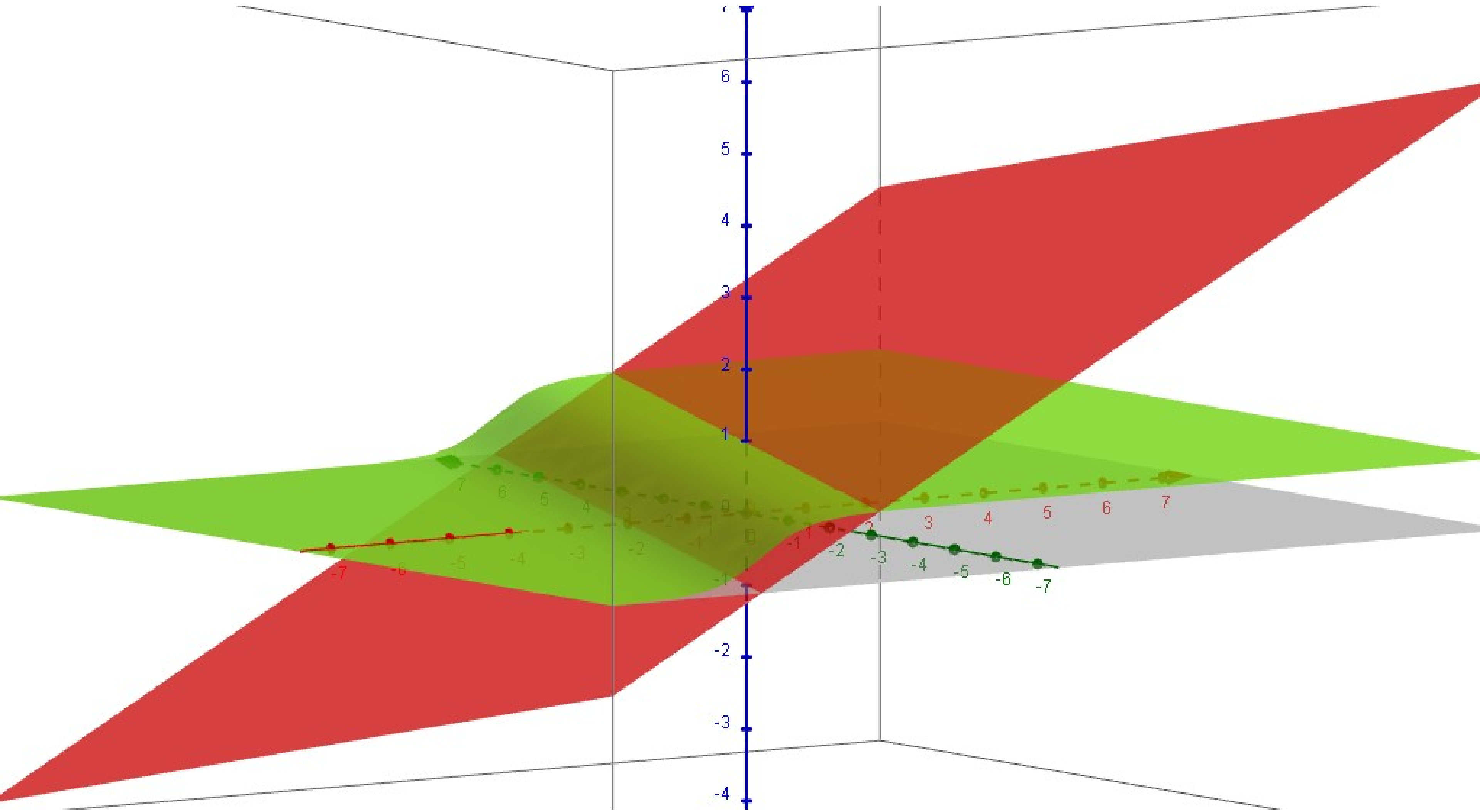
$$\mathbf{C}(\mathbf{w_1};\mathbf{w_2};\mathbf{w_3}) = \sum_{\text{Exemples}} [\mathbf{H}(\mathbf{w_1}\mathbf{x} + \mathbf{w_2}\mathbf{y} + \mathbf{w_3}) - \mathbf{z}]^2$$

soit minimum

Ou (x;y) est un exemple et z la sortie désirée pour cette exemple.

La fonction F:(x;y) $\longrightarrow$ $H(w_1x+w_2y+w_3)$

est une bonne fonction sur les exemples


6
5
4
3
2
1
0
-1
-2
-3
-4
-7
-6
-5
-4
-3
-2
3
4
5
6
7

On cherche à minimiser

$$\mathbf{C}(\mathbf{W}) = \sum_{\textbf{Exemples}} [\mathbf{H}(\mathbf{W} \cdot \mathbf{X}) - \mathbf{z}]^2$$

où $W = (w_1; w_2; w_3)$, $X = (x; y; 1)$ et z est la sortie attendu pour X

Principe de l'algorithme

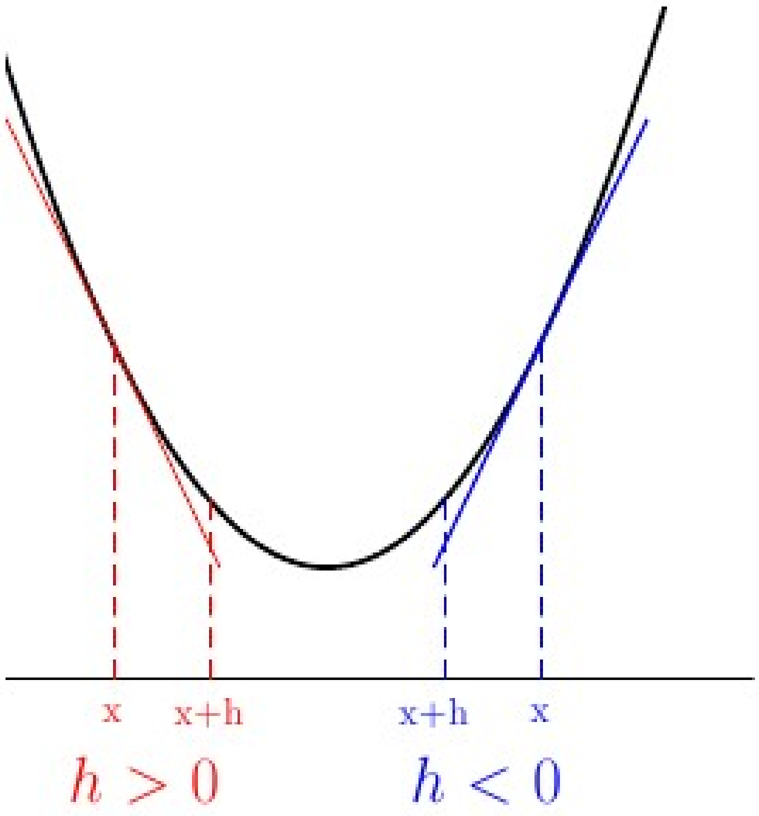
1. On prend des poids W au hasard
2. Tant que le minimum de $C(W)$ n'est pas atteint, faire :
   (a) On prend un exemple X au hasard
   (b) On modifie les poids (pour diminuer $[H(W \cdot X) - z]^2$)

$$W \longleftarrow W - \eta \nabla [H(W \cdot X) - z]^2$$

Où ∇f est le gradient de $f(x_1; \ldots; x_n)$

$$\nabla f = (\frac{\delta f}{\delta x_1}; \ldots; \frac{\delta f}{\delta x_n})$$

Comment modifier les poids : la descente de gradiant



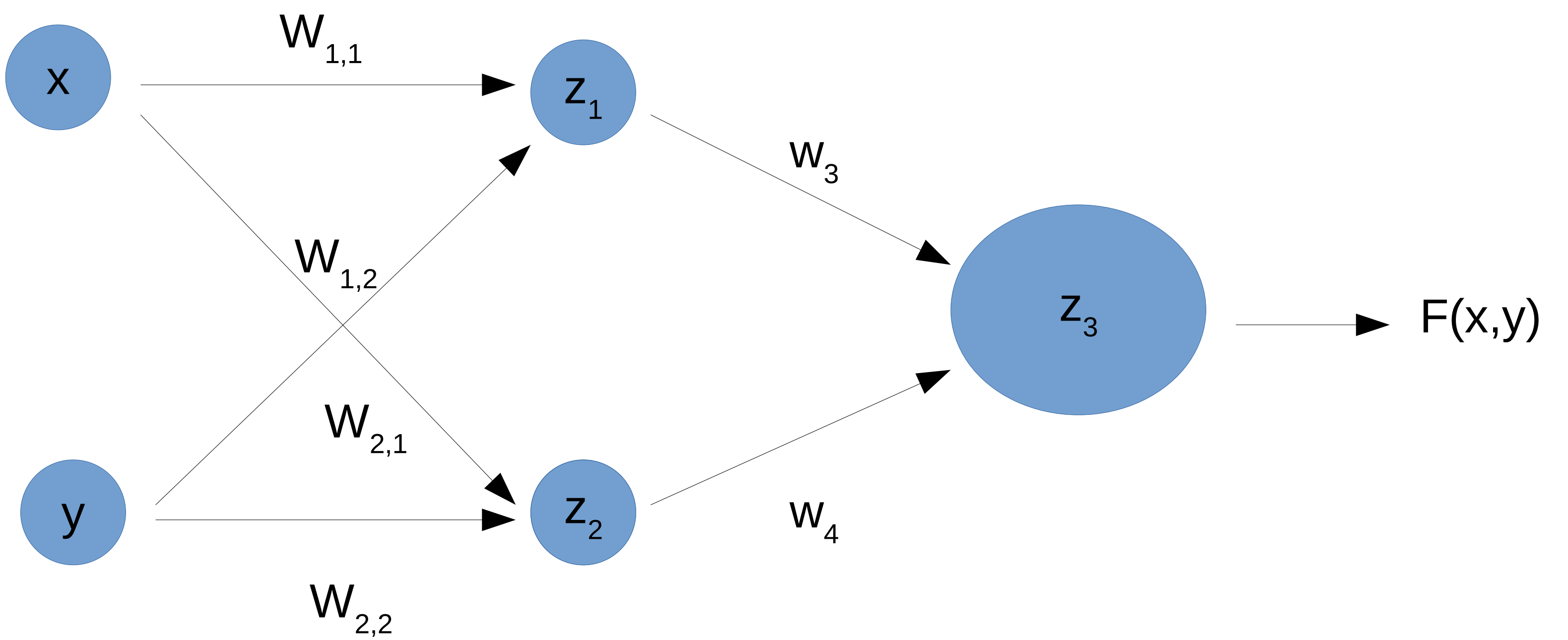
$$f(x+h) \simeq f(x) + hf'(x)$$

Si $h = -\eta f'(x)$, pas $\eta > 0$

alors $f(x+h) \simeq f(x) - \eta[f'(x)]^2$

Si $f'(x) \neq 0$, $f(x+h) < f(x)$

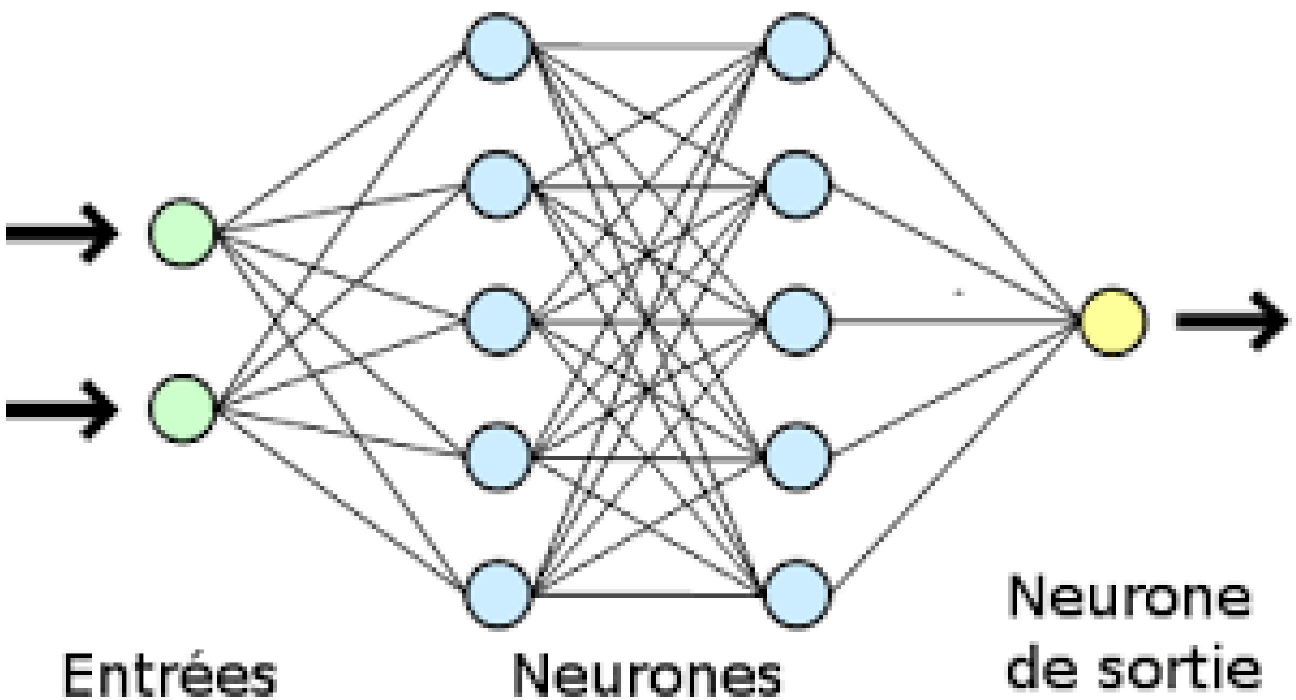
UN RESEAU DE NEURONE



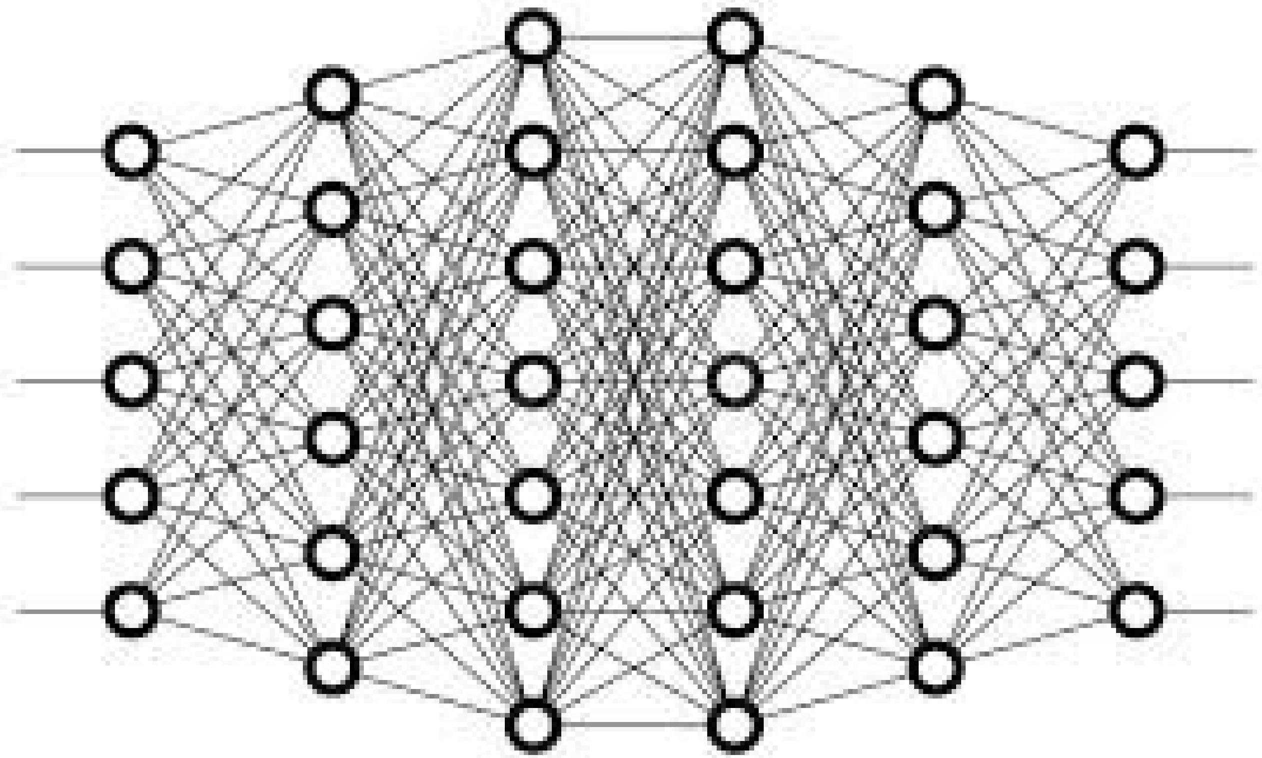
$$\mathbf{z_i = H(w_{i,1}x + w_{i,2}y + w_{i,3})}$$

Chaque choix des poids donne une nouvelle fonction : il faut donc choisir les poids

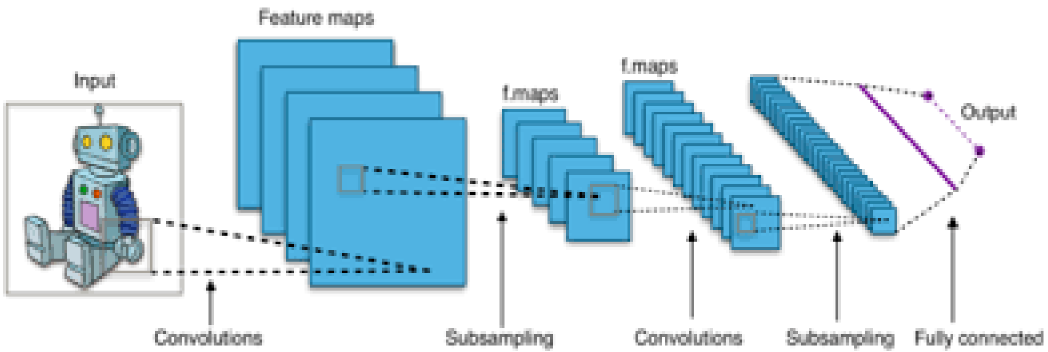
UN RESEAU PLUS PROFOND



ENCORE PLUS GROS

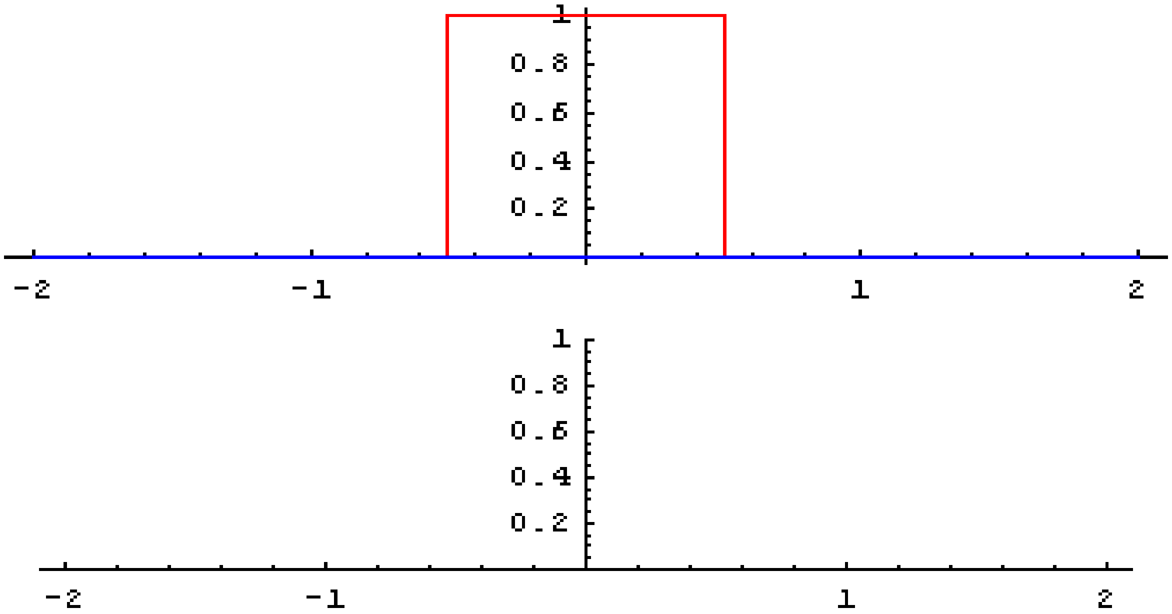


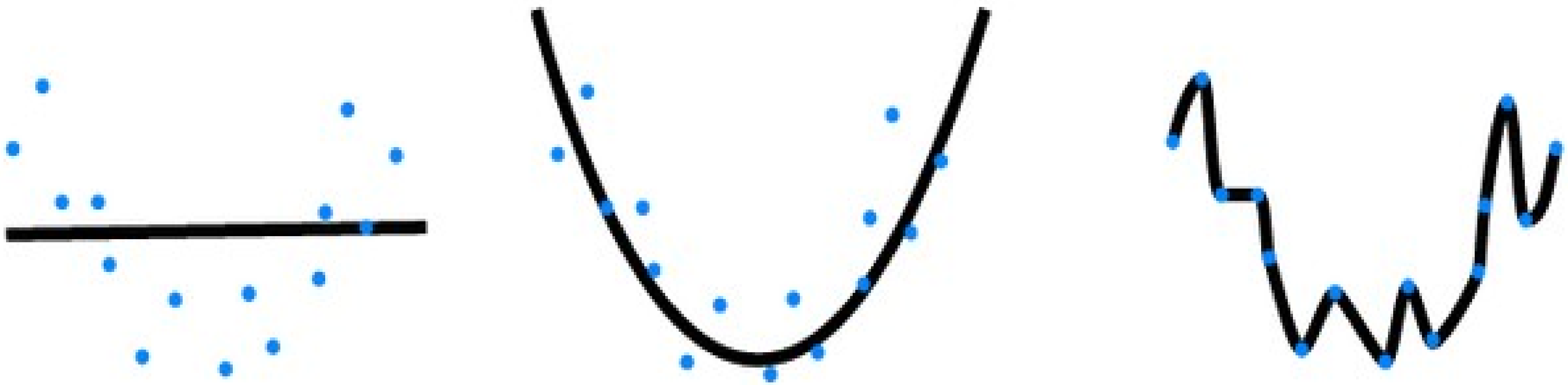
RESEAUX CONVOLUTIFS



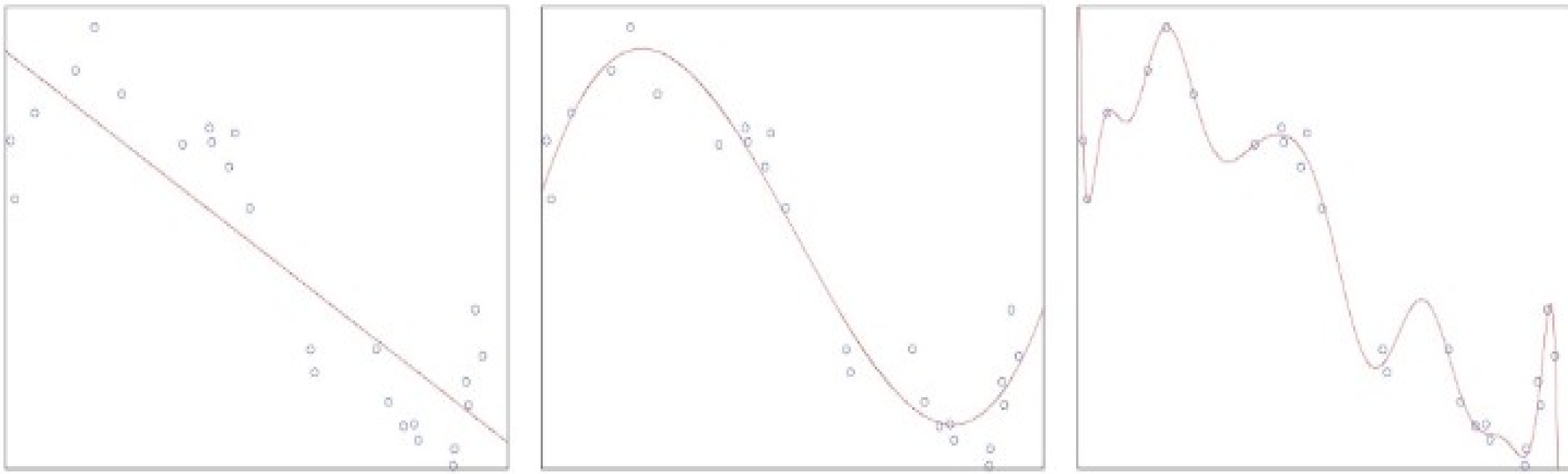
La convolution

$$(f * g)(x) = \int_{-\infty}^{+\infty} f(x-t)g(t)\,\mathrm{d}t$$





Underfitting Desired Overfitting



underfit (degree = 1) | ideal fit (degree = 3) | overfit (degree = 20)