

---

**Module : Apprentissage Automatique**

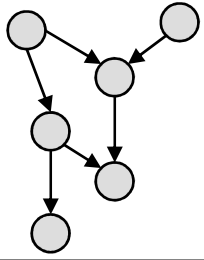
*LES RESEAUX DE NEURONES  
ARTIFICIELS*

Mr. Lakhmissi CHERROUN

Département de Mathématique et Informatique

Université de Djelfa

2011-2012



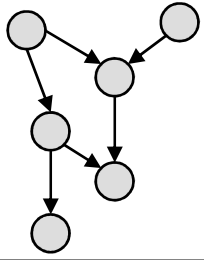
# Introduction

Définition

Contexte Scientifique

Historique

Fondements Biologiques



# Définition

---

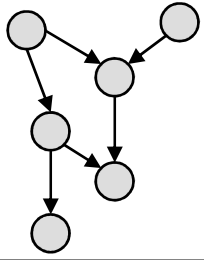
Les réseaux de neurones artificiels sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle.

Chaque processeur élémentaire (neurone artificiel) calcule une sortie unique sur la base des informations qu'il reçoit.

Parallel Distributed Processing :

- Calculs élémentaires et parallèles
- Données/informations distribuées dans le réseau

Inspiration naturelle : analogie avec le cerveau



# Contexte Scientifique

---

## Neuromimétisme et sciences de la cognition :

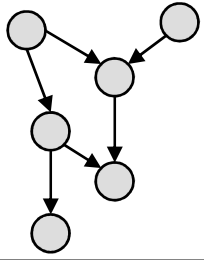
comprendre et simuler le fonctionnement du cerveau  
reproduire les phénomènes cognitifs (I.A.)

## Connexionisme :

outils d'ingénierie performants

## Intelligence computationnelle :

une intelligence basée sur le calcul numérique  
opposée à l'intelligence artificielle (calcul symbolique)  
réseau de neurones; logique floue; algorithmes génétiques; ...



# Historique

## la préhistoire

---

James [1890] :

mémoire associative

McCulloch & Pitts [1943]

*A logical calculus of the ideas immanent in nervous activities*

neurone formel

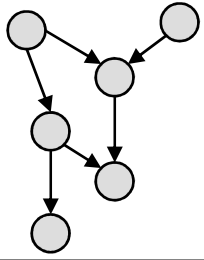
- ⇒ Les ordinateurs à codage binaire (Von Neumann)
- ⇒ L'intelligence artificielle (calcul symbolique)
- ⇒ Les réseaux de neurones

Hebb [1949]

*Organisation of behavior*

le conditionnement est une propriété des neurones

loi d'apprentissage



# Historique

## les premiers succès

---

### Rosenblatt [1957] :

le perceptron, premier modèle opérationnel  
reconnaissance d'une configuration apprise  
tolérance aux bruits

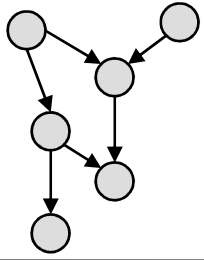
### Widrow [1960] :

adaline, *adaptive linear element*

### Minsky & Papert [1969] :

impossibilité de classer des configurations non linéairement  
séparables

abandon (financier) des recherches sur les RNA



# Historique

## l'ombre et le renouveau

---

[1967 - 1982] :

Mise en sommeil des recherches sur les RNA. Elles continuent sous le couvert de domaines divers.

Grossberg, Kohonen, Anderson, ...

Hopfield [1982] :

modèle des verres de spins

Boltzmann [1983] :

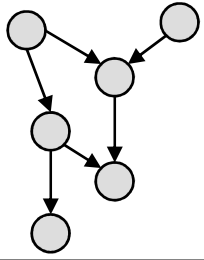
première réponse à Minsky et Papert

[1985] :

la rétro-propagation du gradient et le perceptron multicouche

Rumelhart, McClelland, ... [1985] :

le groupe *Parallel Distributed Processing*



# Domaines d'application

---

- **Classification :**

- répartir en plusieurs classes des objets

- données quantitatives → informations qualitatives

- reconnaissance des formes

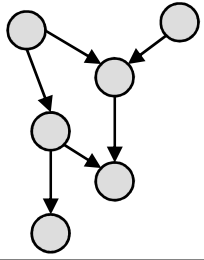
- **Recherche Opérationnelle**

- résoudre des problèmes dont on ne connaît pas la solution

- **Mémoire Associative**

- restituer une donnée à partir d'informations incomplètes et/ou bruitées.





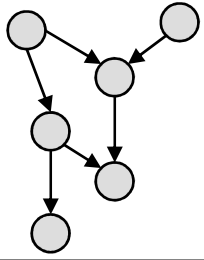
# Fondements Biologiques

## Structure des neurones

Le système nerveux est composé de  $10^{12}$  neurones interconnectés. Bien qu'il existe une grande diversité de neurones, ils fonctionnent tous sur le même schéma.

Ils se décomposent en trois régions principales :

- Le corps cellulaire
- Les dendrites
- L'axone

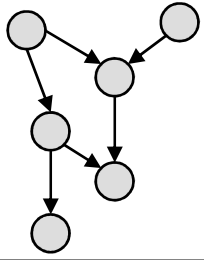


# Fondements Biologiques

## Fonctionnement des neurones

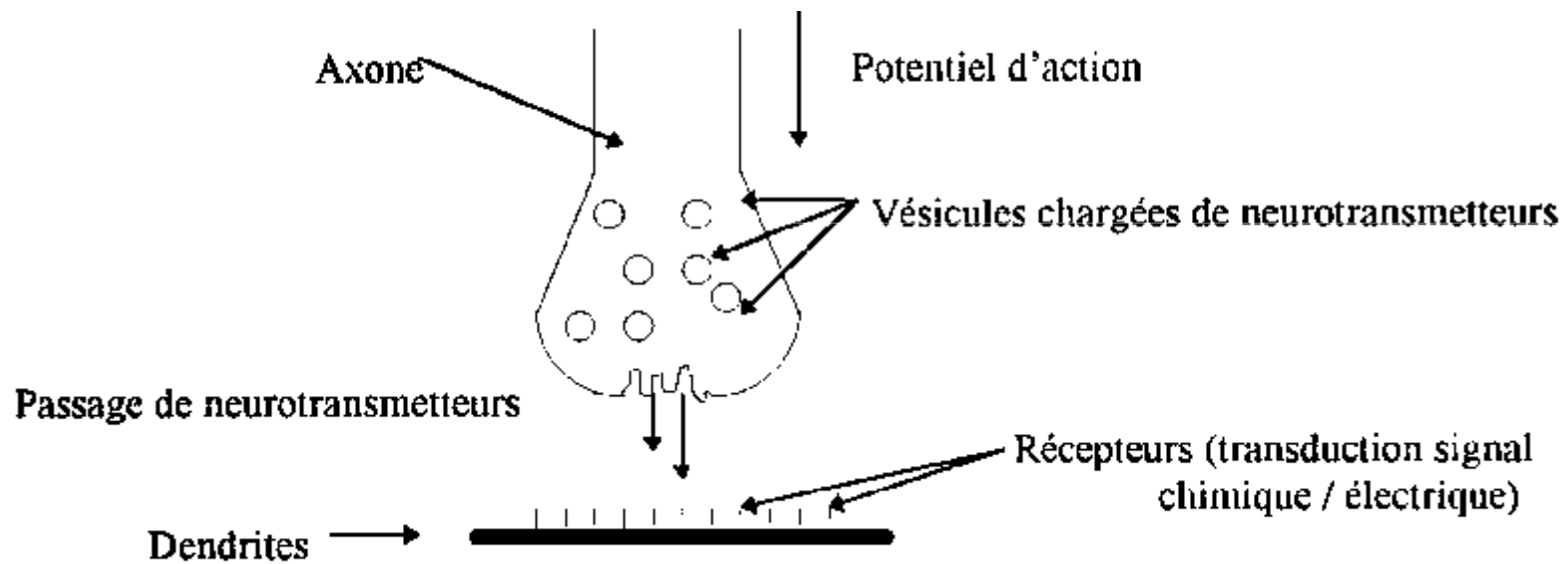
L'influx nerveux est assimilable à un signal électrique se propageant comme ceci :

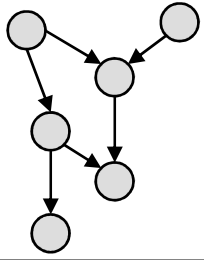
- Les dendrites reçoivent l'influx nerveux d'autres neurones.
- Le neurone évalue l'ensemble de la stimulation reçue.
- Si elle est suffisante, il est excité : il transmet un signal (0/1) le long de l'axone.
- L'excitation est propagée jusqu'aux autres neurones qui y sont connectés via les synapses.



# Fondements Biologiques

## Fonctionnement des neurones





# Fondements biologiques

## Le cerveau

---

Trois couches successives :

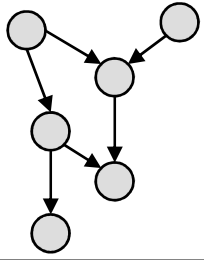
- Le cerveau reptilien (la couche la plus ancienne)
- L'hippocampe (cerveau archaïque)
- Le cortex (la couche la plus récente)

Découpage en régions :

au niveau morphologique et fonctionnel

Adaptation : renforcement de l'efficacité synaptique:

renforcement des corrélations (loi de Hebb)



---

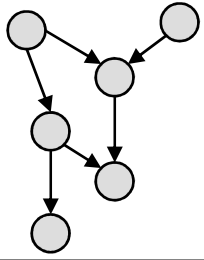
# Les modèles Mathématiques

Le neurone de McCulloch & Pitts

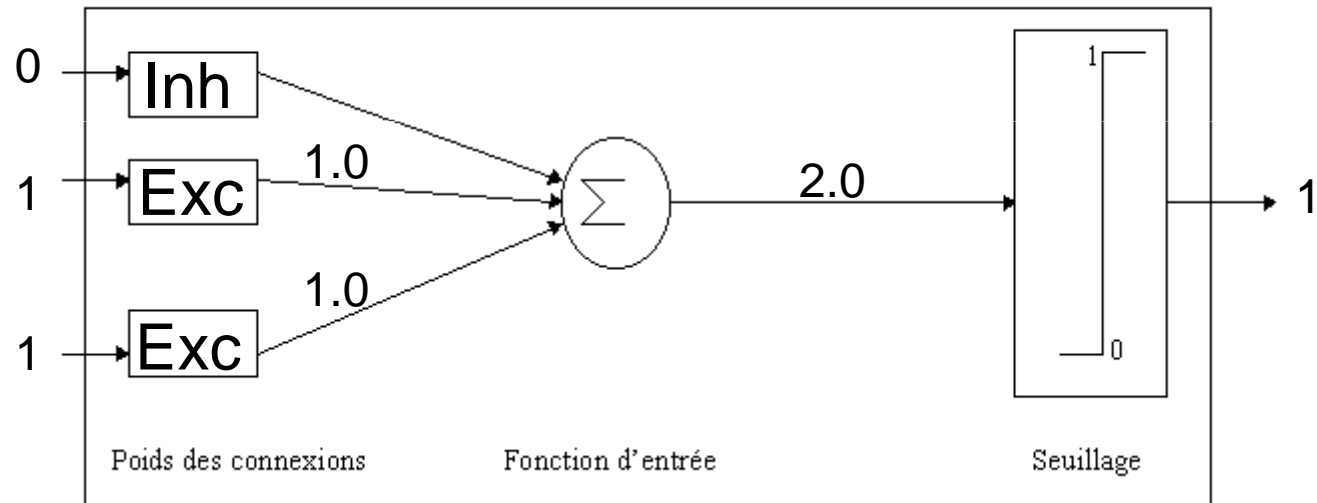
Le neurone formel

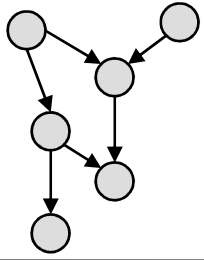
Architecture générale d'un RNA

Structure d'Interconnexion

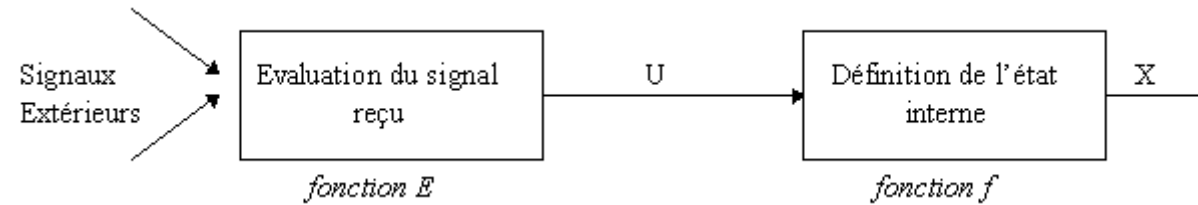


# Le Neurone de McCulloch & Pitts





# Le Neurone Formel

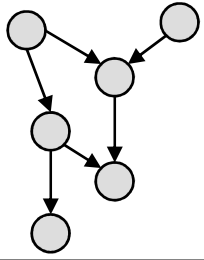


Le neurone formel, l'unité élémentaire d'un RNA, se compose de deux parties :

- évaluation de la stimulation reçue (fonction  $E$ )
- évaluation de son activation (fonction  $f$ )

Il est caractérisé par :

- son état  $X$  (binaire, discret, continu)
- le niveau d'activation reçu en entrée  $U$  (continu)
- le poids des connections en entrée



# Le Neurone Formel

---

La fonction d'entrée :

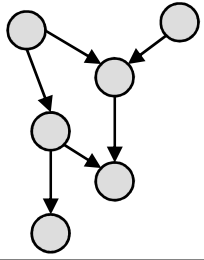
somme pondérée des signaux d'entrée

$$U_i = E(x_1, \dots, x_j, \dots, x_n) = \sum_{j=1}^n W_{ij} x_j$$

Le biais d'entrée (bias input):

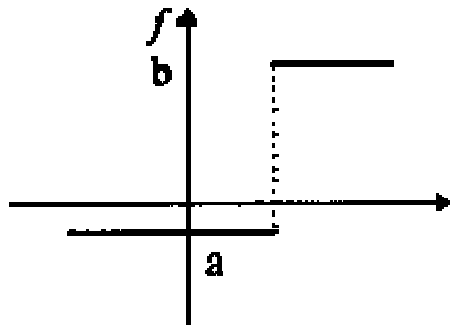
unité fictive dont le poids permet de régler le seuil de déclenchement du neurone



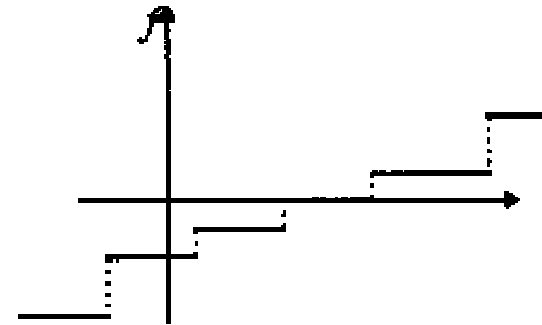


# Le Neurone Formel

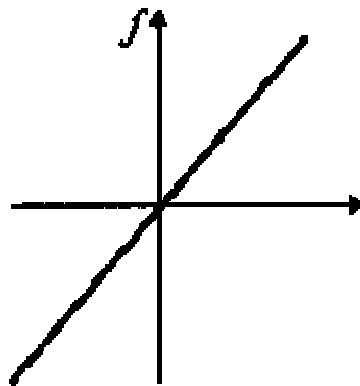
## Les fonctions d'activation :



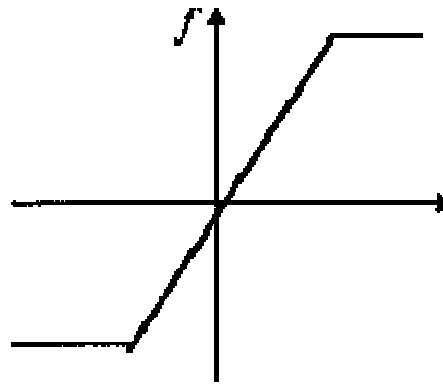
Fonction à seuil



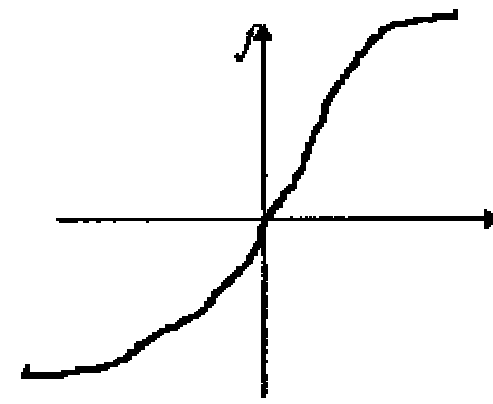
Fonction à valeurs discrètes



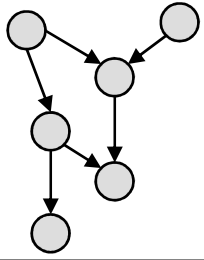
Linéaire



Saturation

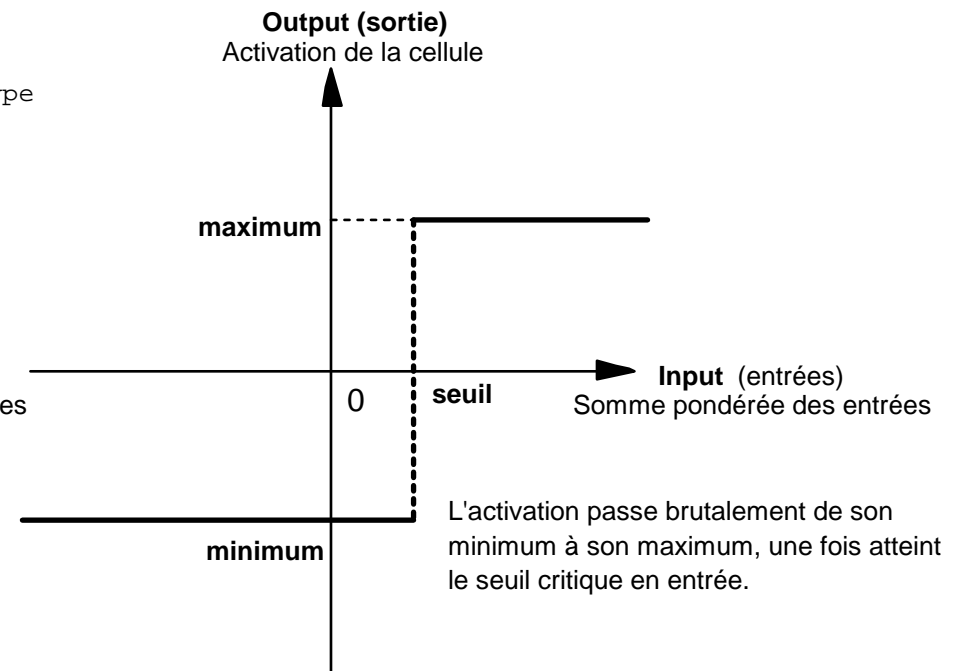
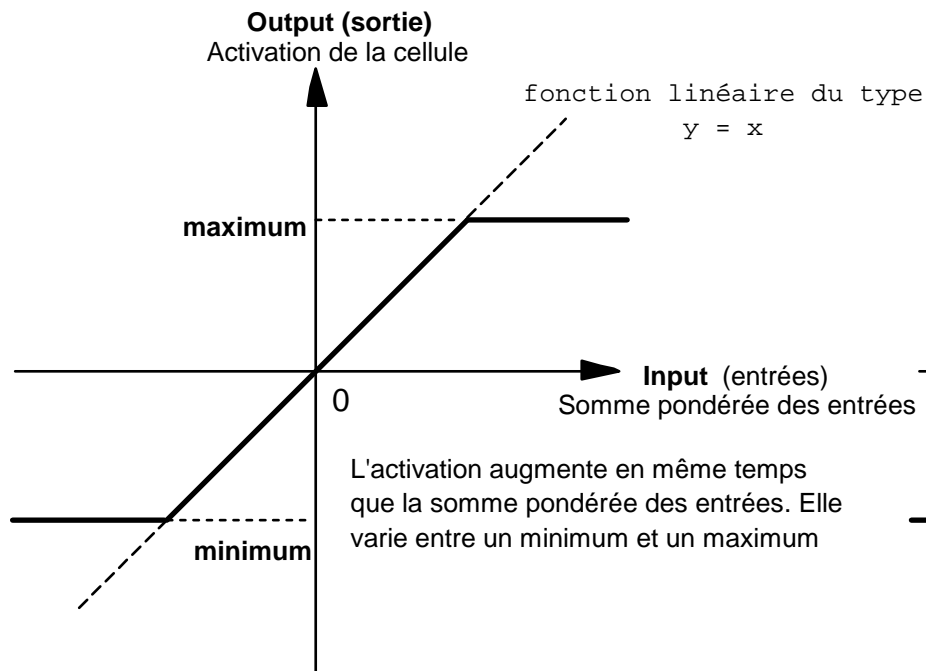


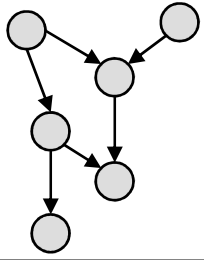
Sigmoïde



# Le Neurone Formel

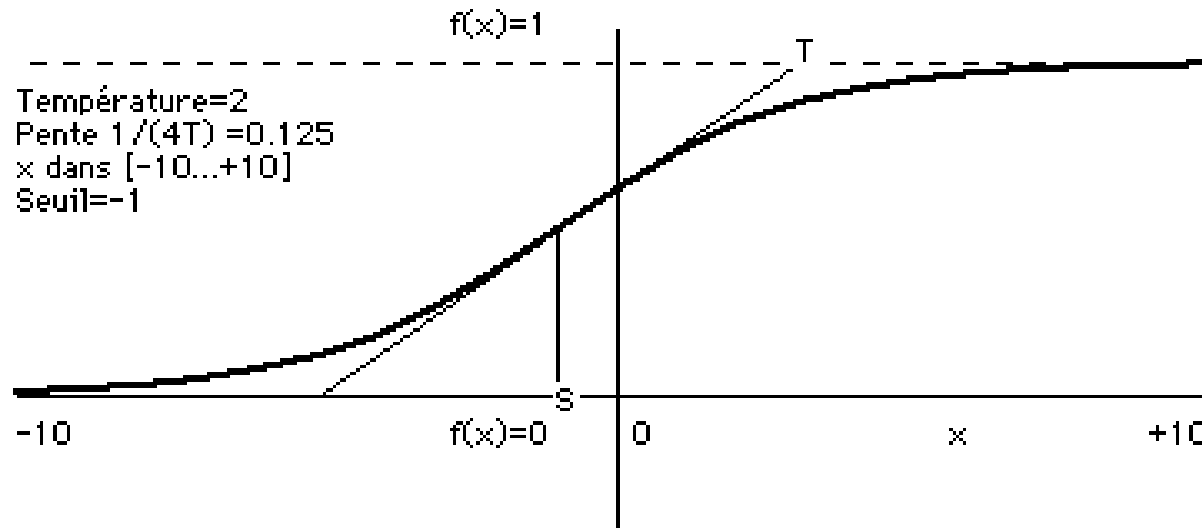
La fonction **linéaire** et la fonction à **seuil** :



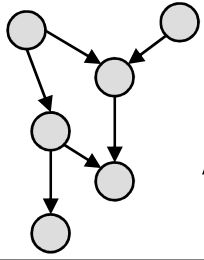


# Le Neurone Formel

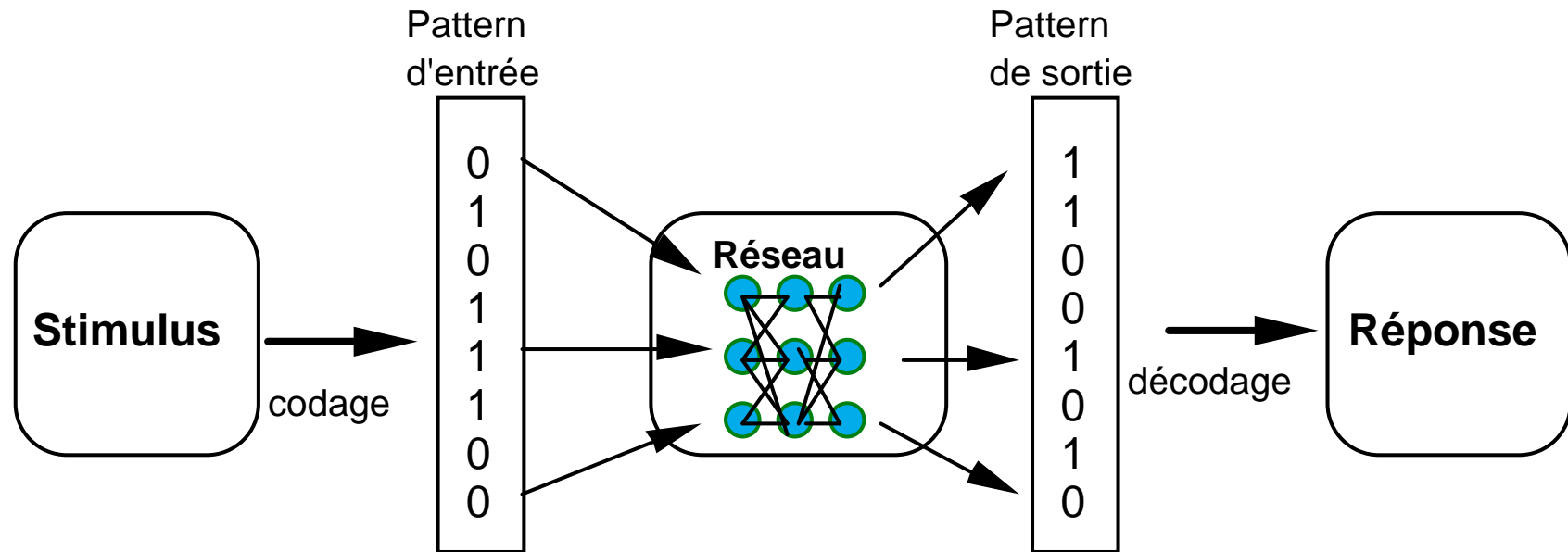
La fonction **sigmoïde** :

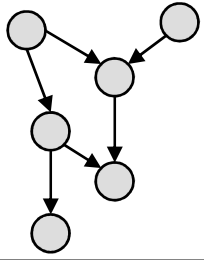


$$a_i^{t+1} = \frac{1}{1 + e^{-\frac{x_i - S_i}{T}}} \text{ avec } x_i = W_{ij} \cdot a_j^t$$



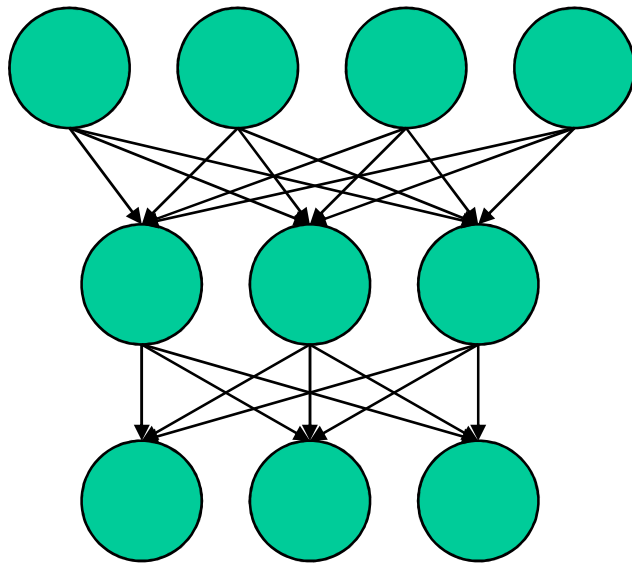
# Architecture générale d'un RNA





# Structure d'Interconnexion

propagation avant (feedforward)

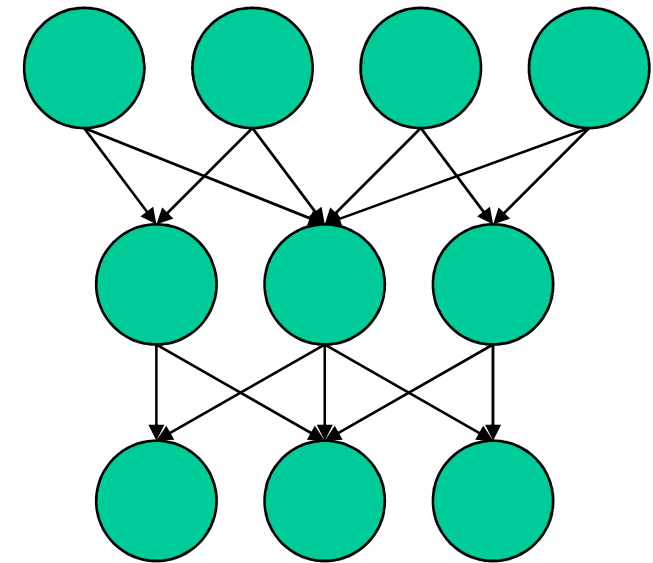


couche d'entrée

couche cachée

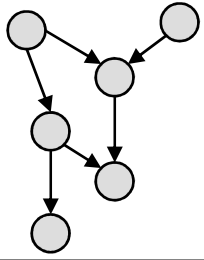
couche de sortie

réseau multicouche



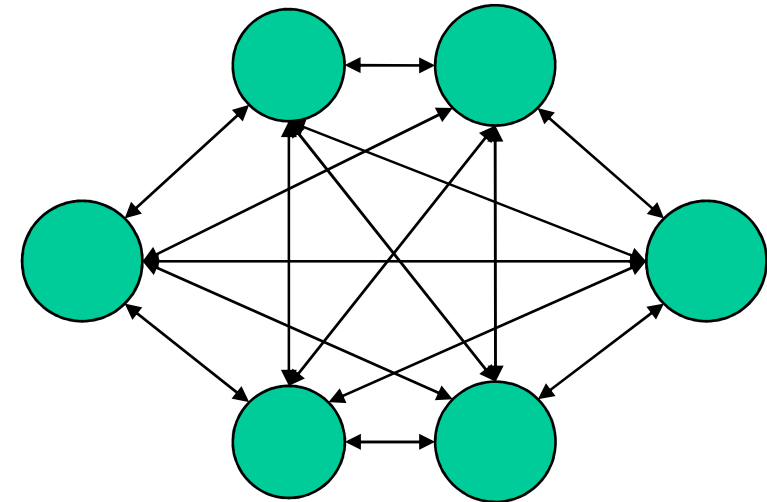
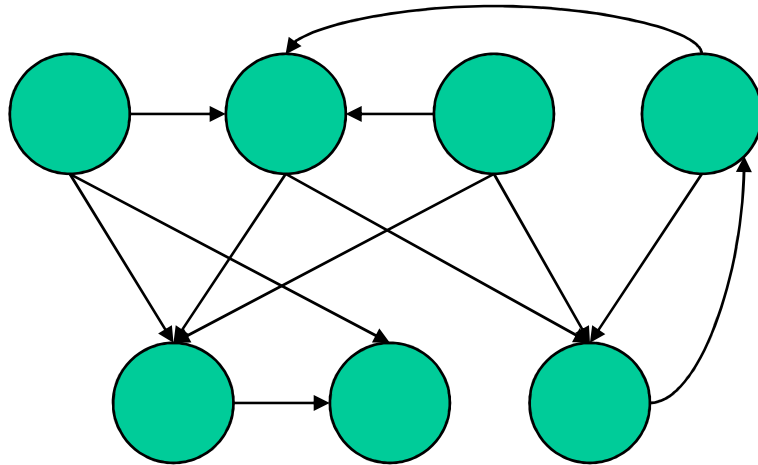
réseau à connexions  
locales

propagation des activations : de l'entrée vers la sortie



# Structure d'Interconnexion

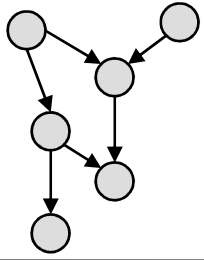
modèle récurrent (feedback network)



propagation des activations :

synchrone : toutes les unités sont mises à jour simultanément

asynchrone : les unités sont mises à jours séquentiellement



---

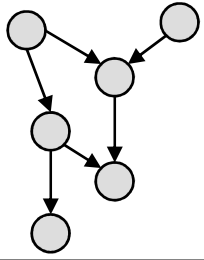
# Apprentissage

## Définition

Apprentissage supervisé

Apprentissage non supervisé

Règles d'apprentissage



# Définition

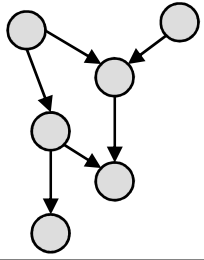
---

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré.

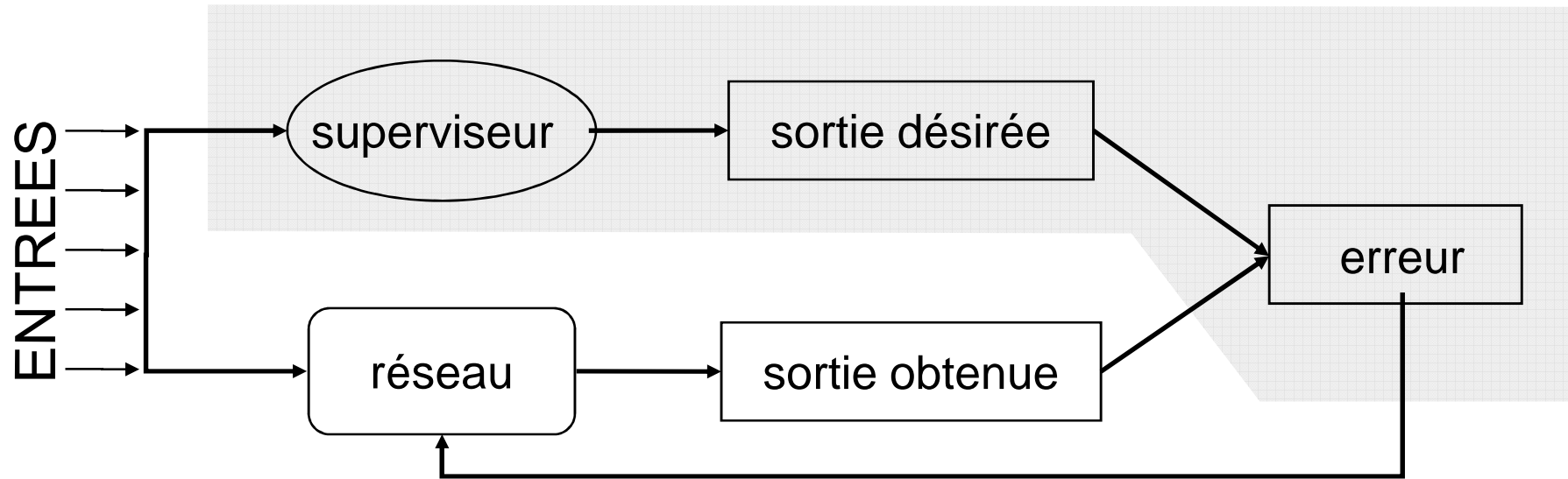
On distingue deux grandes classes d'algorithmes d'apprentissage :

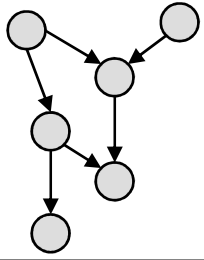
- L'apprentissage supervisé
- L'apprentissage non supervisé





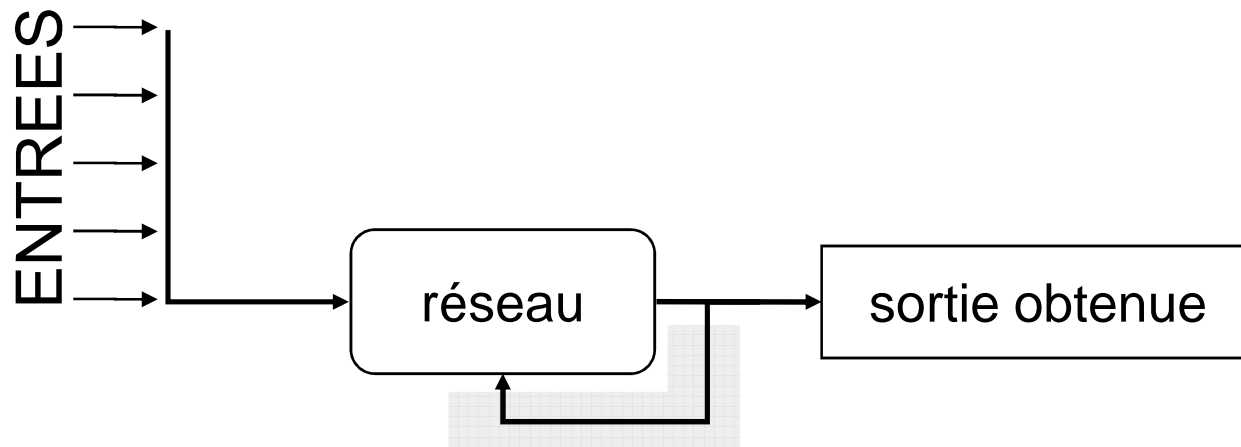
# Apprentissage supervisé

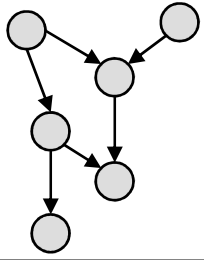




# Apprentissage non supervisé

---

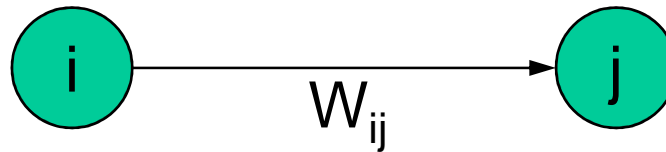




# Règles d'apprentissage

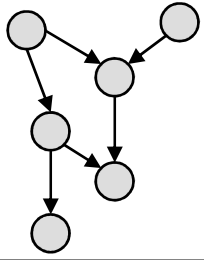
---

L'apprentissage consiste à **modifier le poids des connections** entre les neurones.



Il existe plusieurs règles de modification :

- Loi de Hebb :  $\Delta w_{ij} = R a_i a_j$
- Règle de Widrow-Hoff (delta rule) :  $\Delta w_{ij} = R(d_i - a_i) a_j$
- Règle de Grossberg :  $\Delta w_{ij} = R(a_j - w_{ij}) a_i$

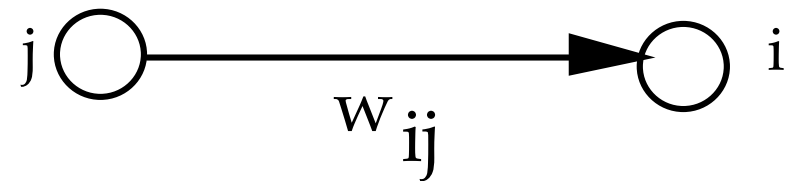


# Règles d'apprentissage

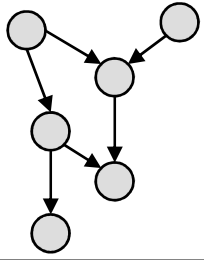
## Loi de Hebb :

Si deux unités connectées sont actives simultanément, le poids de leur connexion est augmenté ou diminué.  $R$  est une constante positive qui représente la force d'apprentissage (learning rate).

	$a_i = -1$	$a_i = 1$
$a_j = -1$	$\Delta W_{ij} = R$	$\Delta W_{ij} = -R$
$a_j = 1$	$\Delta W_{ij} = -R$	$\Delta W_{ij} = R$



$$\Delta W_{ij} = R a_i a_j$$



# Règles d'apprentissage

## Loi de Widrow-Hoff (delta rule) :

$a_i$  activation produite par le réseau

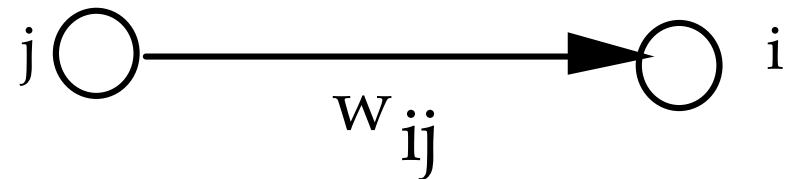
$d_i$  réponse désirée par l'expert humain

Par exemple si la sortie est inférieure à la réponse désirée, il va falloir augmenter le poids de la connexion à condition bien sûr que l'unité  $j$  soit excitatrice (égale à 1). On est dans l'hypothèse d'unités booléennes  $\{0,1\}$ .

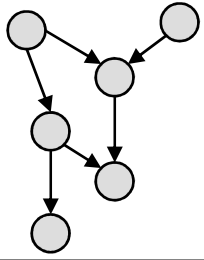
$a_i = 0$	$a_i = 1$
-----------	-----------

$d_i = 0$	$\Delta W_{ij} = 0$	$\Delta W_{ij} = -R$
-----------	---------------------	----------------------

$d_i = 1$	$\Delta W_{ij} = R$	$\Delta W_{ij} = 0$
-----------	---------------------	---------------------



$$\Delta W_{ij} = R(d_i - a_i)a_j$$

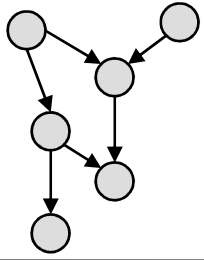


# Règles d'apprentissage

## Loi de Grossberg :

On augmente les poids qui entrent sur l'unité gagnante  $a_i$  s'ils sont trop faibles, pour les rapprocher du vecteur d'entrée  $a_j$ . C'est la règle d'apprentissage utilisée dans les cartes auto-organisatrices de Kohonen

$$\Delta W_{ij} = R a_i (a_j - W_{ij})$$



---

# Différents modèles

Le perceptron

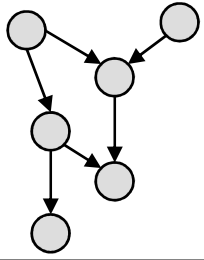
Limite du perceptron

Le perceptron multicouche

Le modèle de Hopfield

Le modèle d'Elmann

Les réseaux ART



# Le perceptron

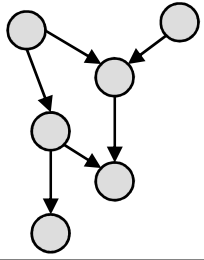
---

Le perceptron de Rosenblatt (1957) est le premier RNA opérationnel. C'est un réseau à propagation avant avec seulement deux couches (**entrée** et **sortie**) entièrement interconnectées.

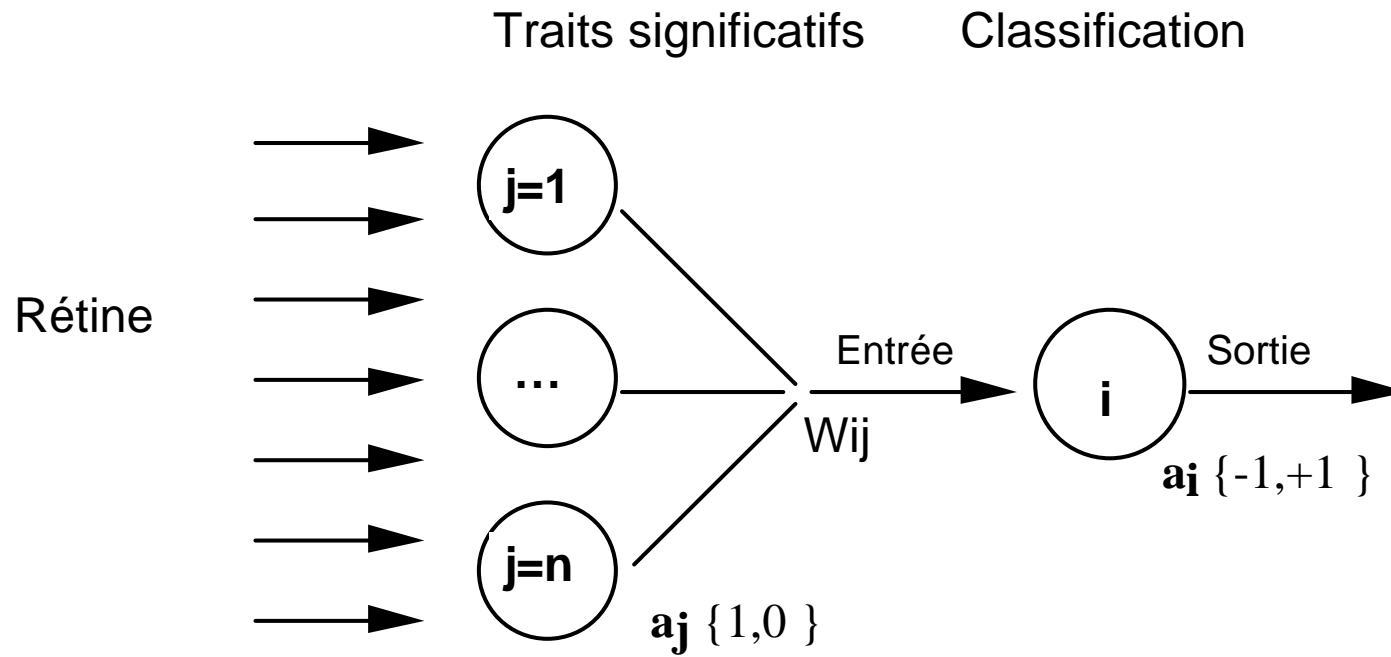
Il est composé de neurones à seuil.

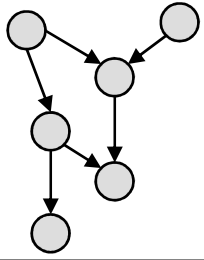
L'apprentissage est supervisé et les poids sont modifiés selon la règle delta.





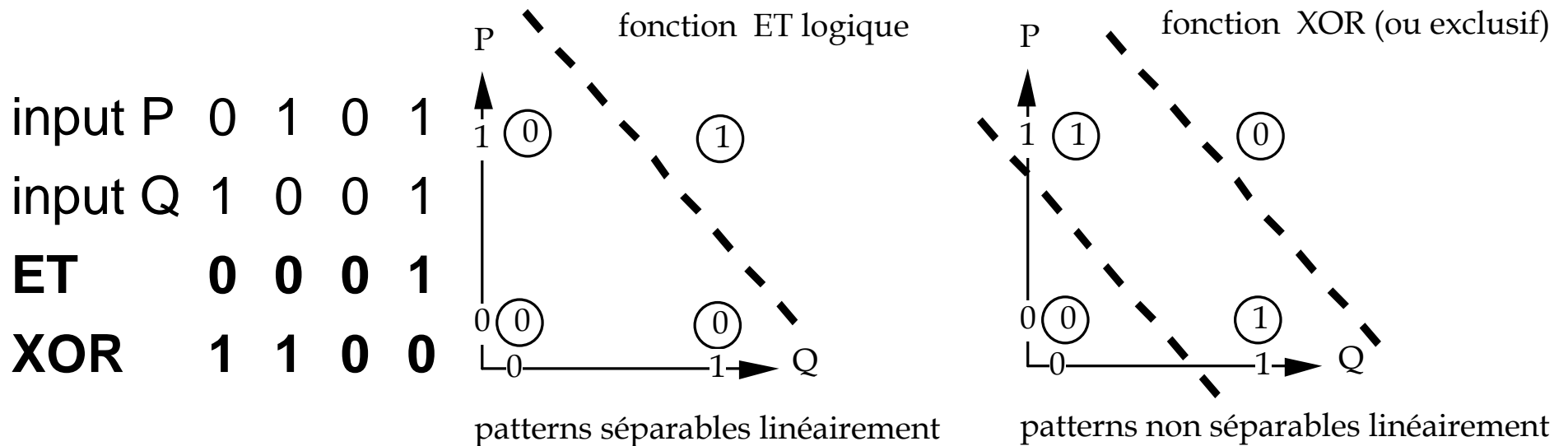
# Le perceptron

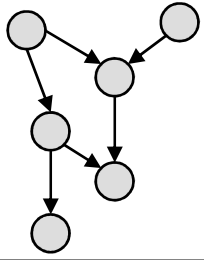




# Limite du perceptron

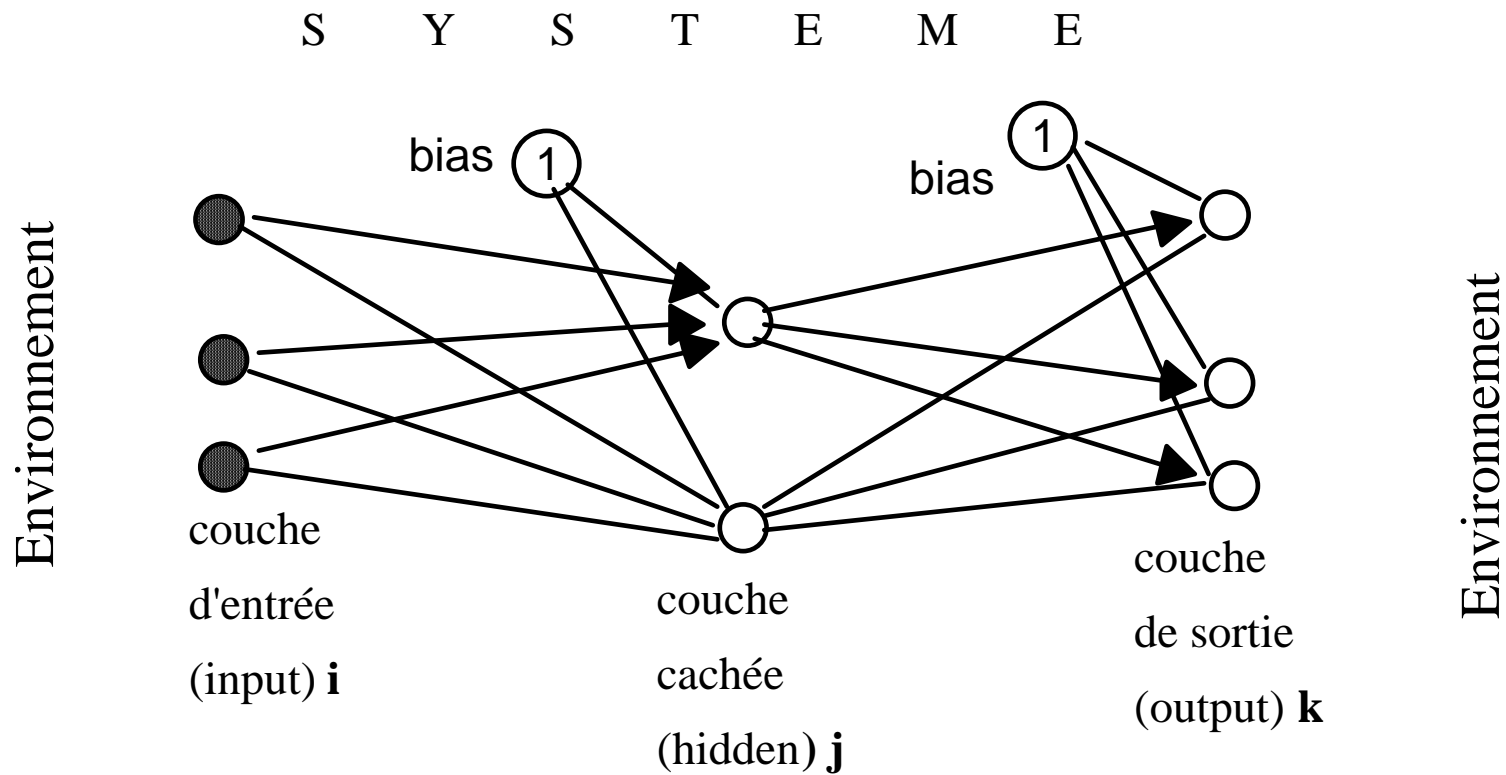
Le perceptron est incapable de distinguer les patterns non séparables linéairement [Minsky 69]

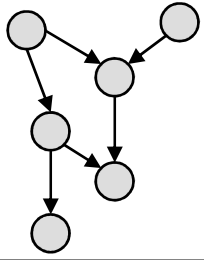




# Le perceptron multicouche

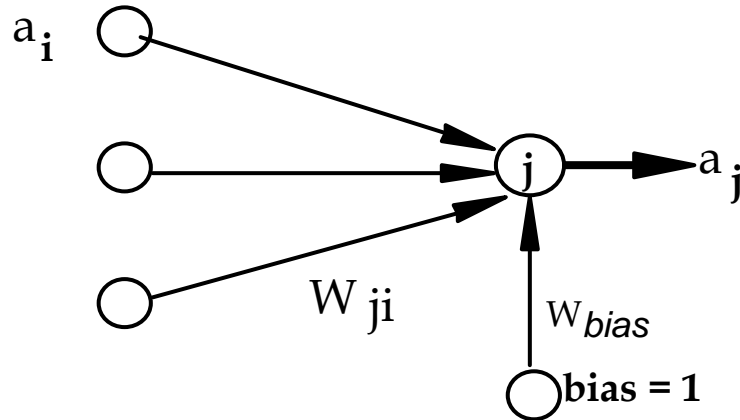
## architecture





# Le perceptron multicouche

activation



$$x_j = \sum w_{ji} a_i$$

$$a_j = f(x_j)$$

fonction sigmoïde

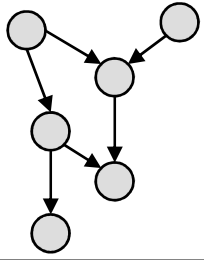
$$a = f(x) = \frac{1}{1 + e^{-x}}$$

$$f'(x) = f(x) \cdot (1 - f(x))$$

fonction tangente hyperbolique

$$a = f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

$$f'(x) = (1 - f(x)) \cdot (1 + f(x))$$



# Le perceptron multicouche

apprentissage : retropropagation de l'erreur

$$S_j = \sum_i a_i W_{ij}$$

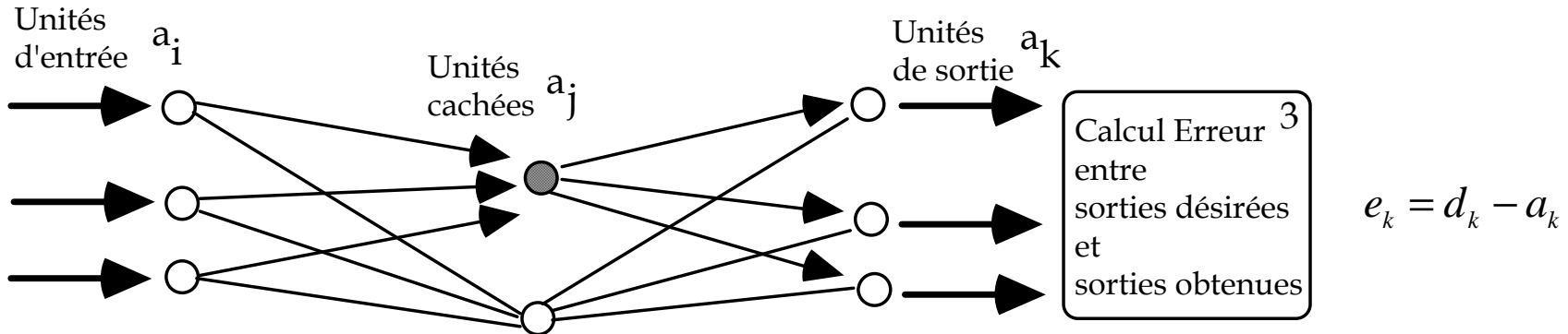
$$a_j = f(S_j)$$

1 Calcul activations  
unités cachées

$$S_k = \sum_j a_j W_{jk}$$

$$a_k = f(S_k)$$

2 Calcul activations  
unités de sortie



$$\delta_j = \left( \sum_k W_{jk} \delta_k \right) \cdot f'(S_j)$$

5 Calcul de l'erreur  
sur les unités cachées

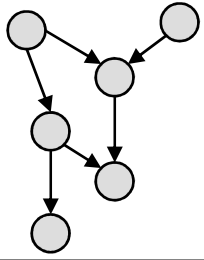
4 Calcul de l'erreur  
sur les unités de sortie

$$\delta_k = e_k \cdot f'(S_k)$$

6 **Ajustement des poids**  
Apprentissage des unités cachées      Apprentissage des unités de sortie

$$\Delta W_{ij} = \epsilon \delta_j a_i$$

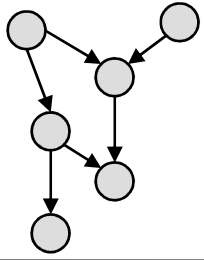
$$\Delta W_{jk} = \epsilon \delta_k a_j$$



# Le modèle de Hopfield

## les mémoires associatives

- Dans une mémoire informatique classique, une information est retrouvée à partir d'une clé arbitraire. Par opposition, une donnée entreposée dans une mémoire associative est accessible à partir d'informations qui lui sont associées.
- La fonction d'une mémoire associative est de restituer une information en tenant compte de sa perturbation ou de son bruit. L'information doit alors se rapprocher d'une information apprise ou connue.
- Si les mémoires associatives restituent des informations qu'elles ont apprises à partir d'entrées incomplètes ou bruitées, il existe aussi des mémoires hétéro-associatives qui en plus peuvent associer plusieurs informations entre elles.



# Le modèle de Hopfield

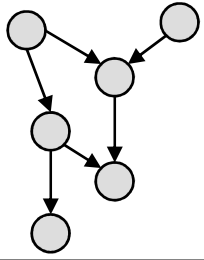
## l'architecture du réseau

Les neurones de Hopfield sont discrets et répondent à une fonction **seuil**. Pour des commodités d'utilisation, on considère une fonction seuil très simple :

$$F(x) = \begin{cases} 1, & \text{si } x > 0 \\ -1, & \text{sinon} \end{cases}$$

Le réseau est complètement connecté, et les connexions sont symétriques.

Les valeurs d'entrée sont binaires (-1, 1) mais peuvent être aisément remplacées par les valeurs binaires usuelles (0, 1) en utilisant une simple transformation.  $A_{(-1,1)} = 2.A_{(0,1)} - 1$



# Le modèle de Hopfield

principe de fonctionnement

**Apprentissage** (loi de Hebb):

$$W_{ij} = W_{ji} = \frac{1}{|P|} \sum_{p \in P} s_i^p s_j^p$$

$$W_{ii} = 0$$

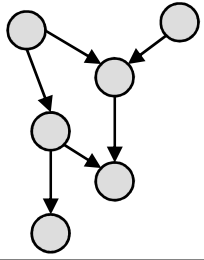
**Utilisation :**

un vecteur est présenté au réseau

les neurones calculent leurs sorties

les sorties sont propagées et on itère jusqu'à la convergence





# Le modèle de Hopfield

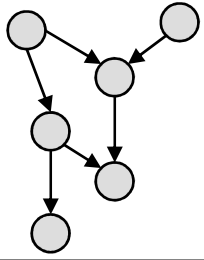
principe de fonctionnement

## Modification de l'état d'un neurone

$$\left\{ \begin{array}{l} \text{si } \sum_{j \in N} W_{ij} \cdot s_j^{t-1} < 0 \text{ alors } s_i^t = -1 \\ \text{si } \sum_{j \in N} W_{ij} \cdot s_j^{t-1} > 0 \text{ alors } s_i^t = 1 \\ \text{si } \sum_{j \in N} W_{ij} \cdot s_j^{t-1} = 0 \text{ alors } s_i^t = s_i^{t-1} \end{array} \right.$$

## Energie du réseau

$$E = -\frac{1}{2} \sum_{i, j \in N} W_{ij} \cdot s_j \cdot s_i$$



# Le modèle de Hopfield

application : reconnaissance de caractère

```

..X.. ..XX. XXXXX XXXXX X.... XXXXX XXXX. XXXXX .XXX. .XXX.
.XXX. ..XX. ...XX ...XX X.... XX... X.... ...X X...X X...X
XX.XX ..XX. ...XX ...XX X.... XX... X.... ...XX X...X X...X
X...X ..XX. ...XX XXXX. X.... XX... X.... ...X. X...X X...X
X...X ..XX. XXXXX ...XX X..X. XXXXX XXXX. ..X.. .XXX. .XXXX
X...X ..XX. XX... ...XX X..X. ...XX X...X .X... X...X ...X
XX.XX ..XX. XX... ...XX XXXXX ...XX X...X .X... X...X ...X
.XXX. ..XX. XX... ...XX ...X. ...XX X...X .X... X...X ...X
..X.. ..XX. XXXXX XXXX. ...X. XXXXX .XXX. .X... .XXX. .XXXX

```

```

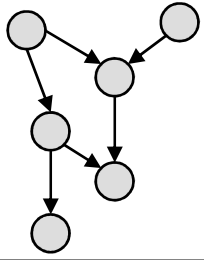
XXX.. ..X.. ..X.. ..X..
XX.X. .XXX. .XXX. .XXX.
XX.XX XX.XX XX.XX XX.XX
X...X X...X X...X X...X
X...X X...X X...X X...X
XX.XX XX.XX XX.XX XX.XX
XXX.. .X.X. .XXX. .XXX.
X.X.. ..X.. ..X.. ..X..

```

```

XXX.. ..X.. ..X.. ..X..
XX.X. XXXX. XXXX. XXXX.
XX.XX XX..X XX..X XX..X
X...X X...X X...X X...X
X...X XX..X XX..X XX..X
X...X X...X X...X X...X
XX.XX XX.XX XX.XX XX.XX
XXX.. XX.X. XX.X. XX.X.
X.X.. ..X.. ..X.. ..X..

```

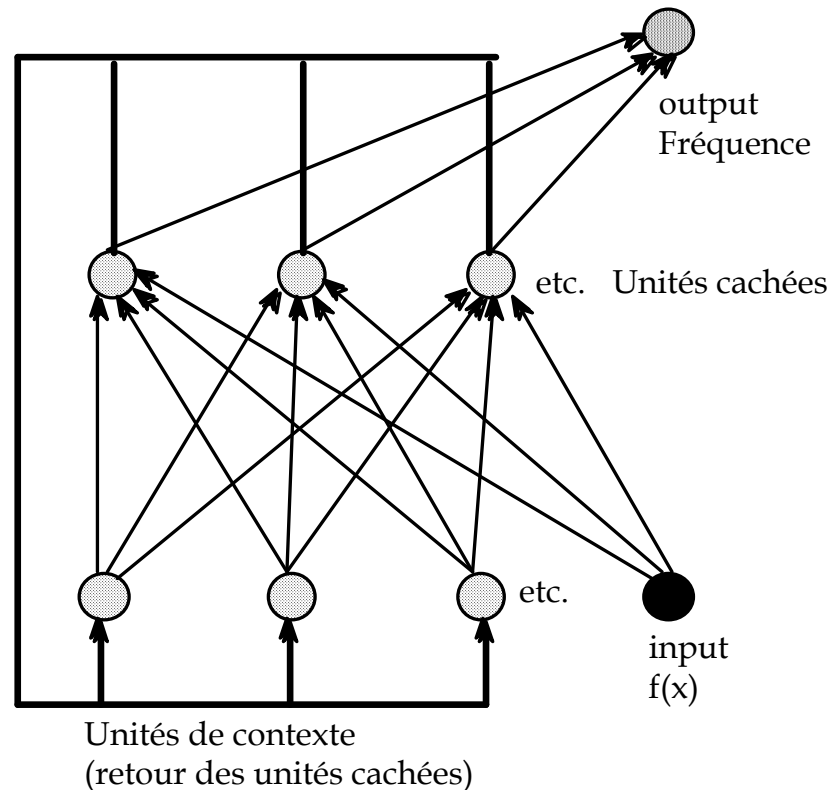


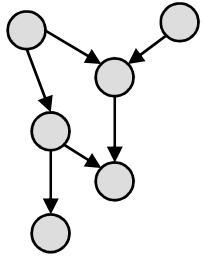
# Le modèle d'Elman

le modèle récurrent de rétro-propagation

**apprentissage** : rétropropagation

**application** : reconnaissance de série temporelle

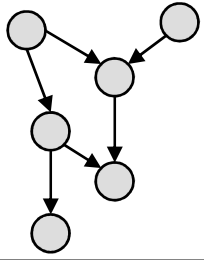




# Le modèle d'Elman

le modèle récurrent de rétro-propagation

- **Phase d'apprentissage**
  - On présente une série temporelle au réseau ( $f(x)=\sin(F \cdot x)$ )
  - La sortie désirée est fixée ( $F$ )
  - Pour chaque élément de la série : rétropropagation
  - On recommence pour deux ou plus valeurs de  $F$
- **Reconnaissance**
  - Lorsqu'on présente une fonction de fréquence apprise, après quelques itérations, le réseau réussit à la reconnaître
- **Généralisation**
  - Lorsqu'on présente une fonction de fréquence non apprise, le réseau peut interpoler cette dernière



# Les réseaux concurrentiels

## présentation

---

Un réseau concurrentiel comporte **une couche** de neurone dite compétitive.

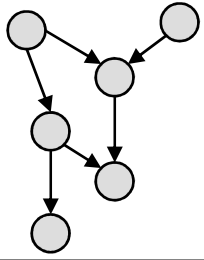
Dans cette couche :

- les neurones réagissent différemment aux entrées

- un neurone est élu vainqueur

- le gagnant à le droit de modifier ses poids de connexion

**Type d'apprentissage** : supervisé / non supervisé



# Les réseaux concurrentiels

## fonctionnement

### Fonction d'activation

$$S_j = \sum_i W_{ij} a_i$$

Si  $S_j$  est le max sur toute la couche compétitive,  $a_j = 1$

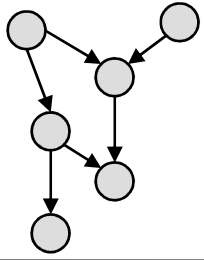
Sinon  $a_j = 0$

### Apprentissage

- La somme des poids arrivant sur une unité compétitive reste constante et égale à 1
- Seule l'unité gagnante a le droit de modifier ses poids, afin qu'ils se rapprochent encore plus du vecteur d'entrée (mais ils restent normalisés)

### Compétition par inhibition

- Il est possible de simuler la compétition par des connections inhibitrices entre les unités compétitives



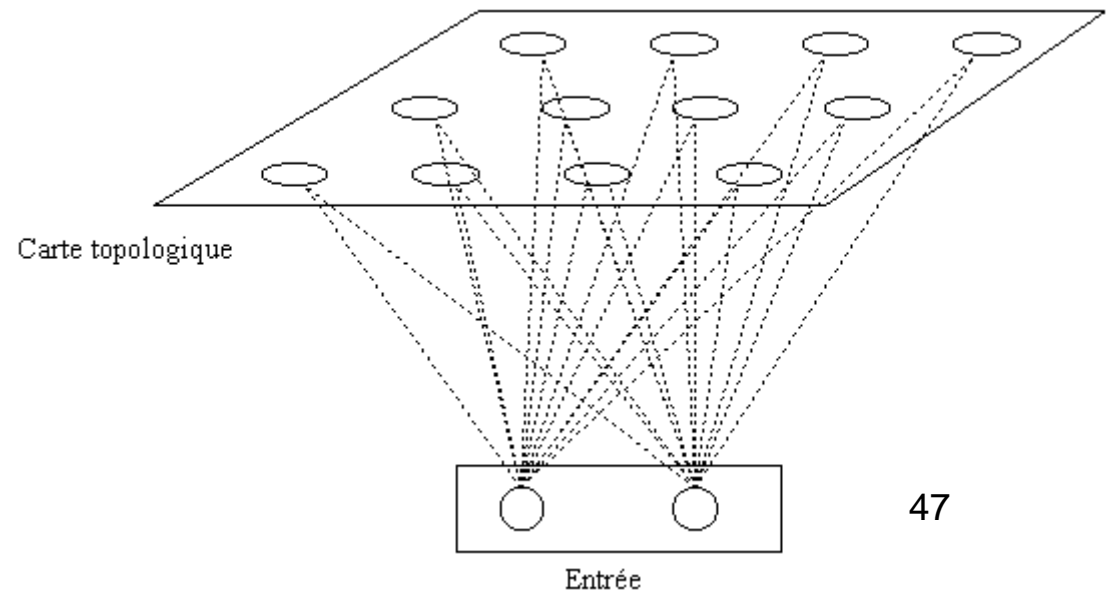
# Les réseaux concurrentiels

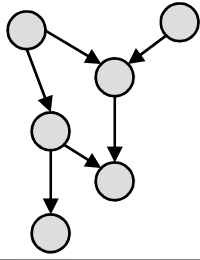
## les cartes topologique de Kohonen

Un réseau de Kohonen est composé d'une couche d'entrée et d'une couche compétitive.

La couche compétitive possède une structure topologique, ce qui permet de définir un voisinage pour le neurone.

L'apprentissage est non supervisé.





# Les réseaux concurrentiels

## les cartes topologique de Kohonen

Les voisins proches du gagnant modifient positivement leurs poids

Les voisins éloignés du gagnant modifient négativement leurs poids

Après l'apprentissage, les poids décrivent la densité et la structure de la répartition des vecteurs d'entrée

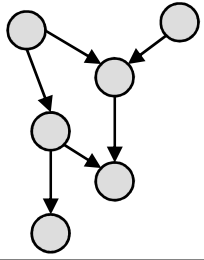
### Application :

Classification non supervisée

Réduction du nombre de dimension

Analyse en composantes principales





# Les réseaux ART

[Carpenter & Grossberg 87]

Les réseaux ART sont utilisés comme détecteurs de caractéristiques pour classer des patterns présentés au système, tels qu'ils arrivent et sans autre information que leur description interne, grâce aux deux couches de neurones F1 et F2 :

F1 qui détecte les caractéristiques des patterns étudiés.

F2 qui classe ces patterns en catégories.

C'est un apprentissage non supervisé qui fait appel simultanément aux deux aspects de la mémoire :

la mémoire à long terme (**LTM**) qui gère les interactions entre les couches F1 et F2 et s'enrichit pendant la phase d'apprentissage.

la mémoire à court terme (**STM**) qui décrit l'état interne des couches F1 et F2 et se modifie à chaque cycle du système. <sup>49</sup>