

EFREI 2011/2012 L3

Aide à la Décision Réseaux de Neurones

Support de cours

Réseaux de neurones

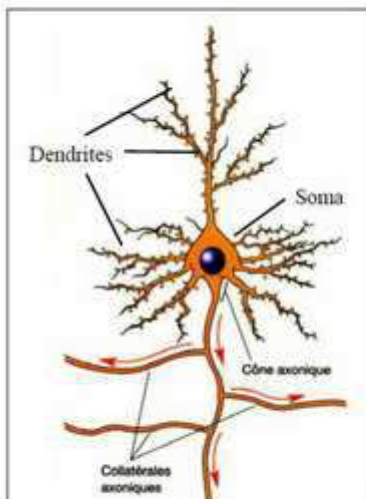
Une introduction

Cours « Aide à la Décision »
EFREI – 2010/2011 – L3

© Hervé BARBOT, 2005-2010 – www.proactitude.com

Hervé Barbot, 2009-2011

1



Diamètre de la cellule: 0.01 - 0.05 mm,
Soma (corps cellulaire)
formes variables (gén. sphériques),
20 μm de diamètre,
contient le noyau,
entourée d'une membrane de 5 nm d'ép.

Axone
unique aux cellules nerveuses,
diamètre: 1-25 μm (humain),
1 mm (poulpe),
longueur: 1 mm to 1 m. (!!!),
connexion vers les autres neurones
(synapses),
permet la transmission d'information.

Dendrites
reçoivent les signaux des autres neurones,
chacune couverte de centaines de
synapses.

Hervé Barbot, 2009-2011

2

- Comment est-ce que cela marche ?
 - Il y a beaucoup d'inconnu sur le fonctionnement du cerveau, sur la façon dont il « apprend » à traiter de l'information
 - Les neurones collectent des signaux venant des autres neurones via les « dendrites »
 - Un neurone envoie des signaux (éléments d'information) aux autres neurones via les « axones »
 - À la terminaison d'un axone, une « synapse » convertit l'activité d'un axone en un signal électrique qui excite ou inhibe l'activité du neurone connecté

Hervé Barbot, 2009-2011

3

- Un cerveau, c'est :
 - ≈ 10^{11} neurones
 - ≈ 10^4 à 10^5 connexions par neurone
 - ≈ 10^{-3} sec par cycle
 - ≈ 10^{-1} sec en moyenne pour une activité cognitive (ex. reconnaissance d'un visage)

→ Environ 100 cycles de traitement !
 ...insuffisant pour une activité complexe

→ Le cerveau doit donc effectuer des opérations en parallèle !!!

Hervé Barbot, 2009-2011

4

- Conditions d' "excitabilité" :
 - Un neurone est considéré comme un semi-conducteur, ou un média "en sens unique" de transfert d'information.
 - La vitesse moyenne de transfert est entre 10 et 120 mètres / seconde.
 - Un signal électrique peut "exciter" un autre neurone si le signal est supérieur à une limite fixée.
 - La réaction d'un neurone peut être positive (activation) ou négative (inhibition)

Hervé Barbot, 2009-2011

5

MODÉLISATION D'UN NEURONE

Hervé Barbot, 2009-2011

6

Modélisation

- Intuition
 - Capacité (intelligence?) = connexions entre les neurones
- Propriétés
 - Unité = neurone
(modélisation biologiquement plausible)
 - Pas de règles explicites d'inférence
 - Pas de « maître de cérémonie »

Hervé Barbot, 2009-2011

7

Neurone formel

- Plusieurs entrées (dendrites) et une sortie (axone)
- Excitation / inhibition = coefficients associé aux entrées = « poids synaptiques »
- Les coefficients sont ajustés au cours d'une phase d'apprentissage
- Forme simple :

$$f\left(\sum_i a_i w_i\right)$$

= la valeur d'une « fonction d'activation » pour une somme pondérée des entrées

Hervé Barbot, 2009-2011

8

• **Modèle binaire**

- Entrées et sorties sont égales à 0 ou 1
- Une somme pondérée des entrées est calculée

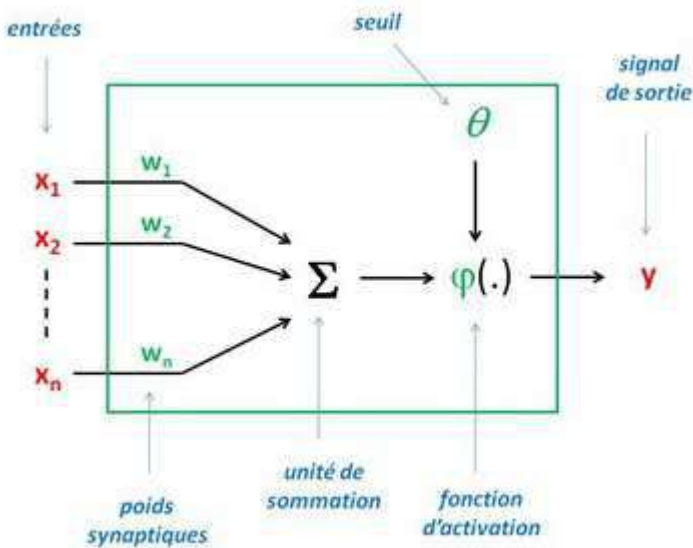
$$\sum_i w_i x_i$$

- Somme > seuil
« activation » : résultat = **1** $\sum_i w_i x_i \geq \theta$

- Somme < seuil
« inhibition » : résultat = **0** $\sum_i w_i x_i < \theta$

Hervé Barbot, 2009-2011

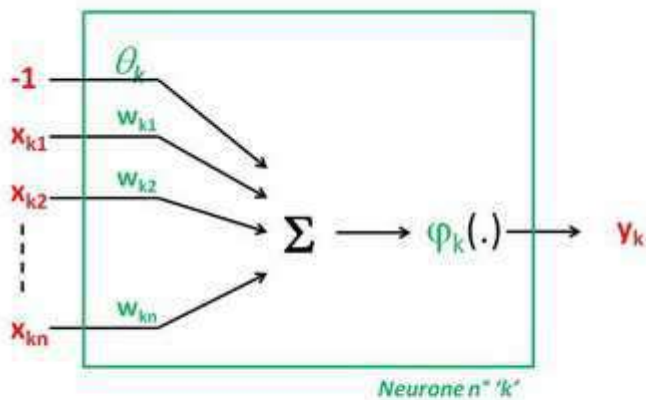
9



Hervé Barbot, 2009-2011

10

Modèle étendu



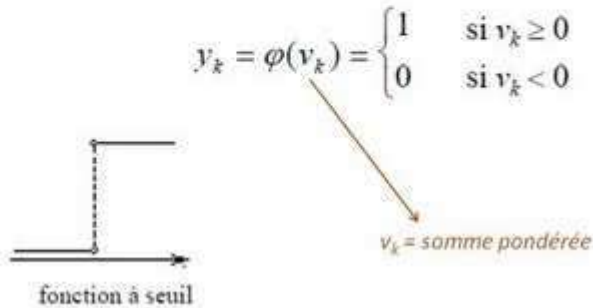
Hervé Barbot, 2009-2011

11

Fonctions d'activation φ

= valeur de sortie d'un neurone

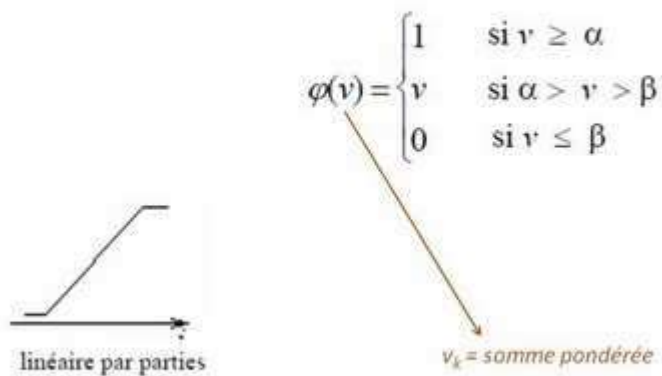
- Fonction à seuil



Hervé Barbot, 2009-2011

12

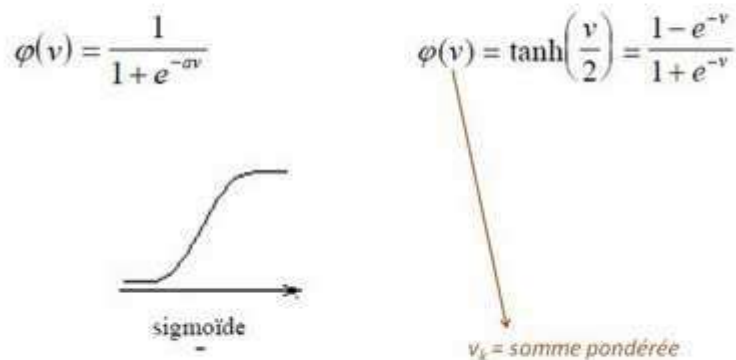
- Fonction linéaire par parties



Hervé Barbot, 2009-2011

13

- Fonction sigmoïde



Hervé Barbot, 2009-2011

14

RÉSEAU DE NEURONE ARTIFICIEL (RNA)

Hervé Barbot, 2009-2011

15

RNA

- Unité de traitement
- Canaux de données en entrées
- Pondération
- Canal de données en sortie

Cerveau

- Noyau / Cellule
- Dendrites
- Synapses
- Axone

Hervé Barbot, 2009-2011

16

- Architecture basé sur celle d'un cerveau
 - Architecture parallèle
- Grand nombre d'éléments inter-connectés
 - Coopération pour la résolution d'un problème
 - Une forme d'ordinateur multi-processeurs
 - Éléments de base simples
 - Haut degré d'interconnexion
 - Messages simples
 - Adaptation de l'interaction entre éléments

Hervé Barbot, 2009-2011

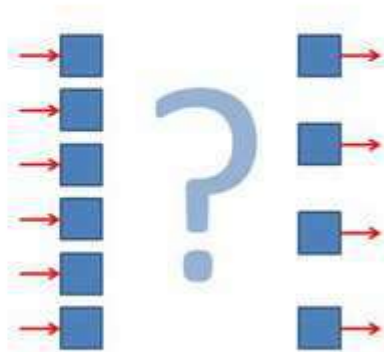
17

- RNA = graphe orienté valué
(≈ automate simple)
 - Noeud = neurone artificiel
 - Arc = transfert d'information d'un neurone à un autre
(= relations sortie -> entrée)
 - Valeur d'un arc = poids synaptique

Hervé Barbot, 2009-2011

18

- Certains neurones reçoivent les informations d'entrée, d'autres fournissent les résultats



Hervé Barbot, 2009-2011

19

Poids synaptiques

- Une matrice représente les poids des liaisons entre les différents neurones
- Activation / inhibition :
 - Activation : poids positif
 - Inhibition : poids négatif

$$W = \begin{bmatrix} W_{11} & \dots & W_{1n} \\ \dots & \dots & \dots \\ W_{m1} & \dots & W_{mn} \end{bmatrix}$$

Matrice d'un RNA
à n neurones

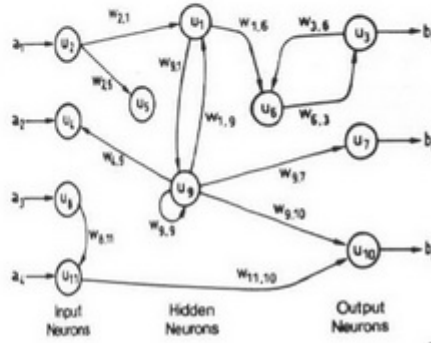
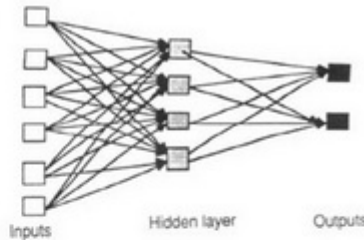
Hervé Barbot, 2009-2011

20

- Maillage complet ou maillage partiel ?

- Dans la pratique, maillage partiel

- Peut être résolu par la mise à '0' de certains poids synaptiques



Hervé Barbot, 2009-2011

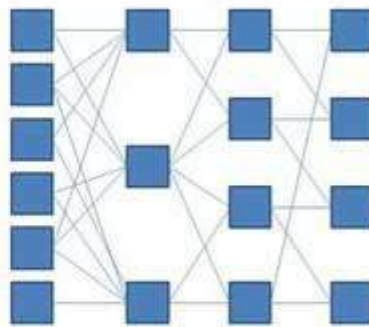
21

Couches de neurones

- Architecture classique (mais pas unique) :

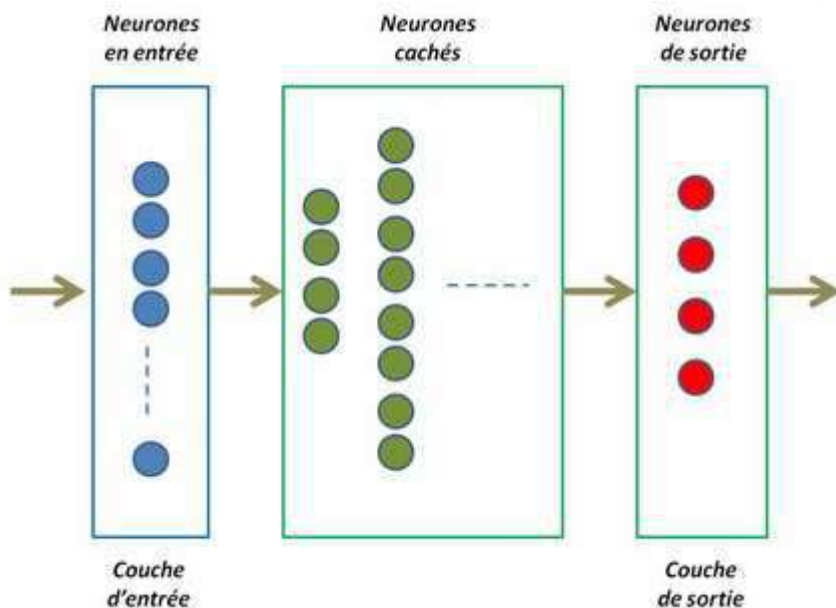
- les neurones sont organisés en couches

- chaque neurone d'une couche N fournit des données aux neurones de couche N+1



Hervé Barbot, 2009-2011

22



Hervé Barbot, 2009-2011

23

Couche d'entrée

- Réception des données initiales
Description du problème à traiter
Transmission au réseau (de neurones)
 - Neurones similaires : même nombre de données reçues, mais traitements spécifiques (pondération)
 - Sorties = abstraction du problème à résoudre

Hervé Barbot, 2009-2011

24

Couches cachées

- Représentent
 - le traitement,
 - la reconnaissance,
 - la résolution
du problème à résoudre
- Pas de "lien" avec l'utilisateur du réseau
- Le nombre de couches nécessaires peut ne pas être précisément défini au début de l'utilisation du réseau

Hervé Barbot, 2009-2011

25

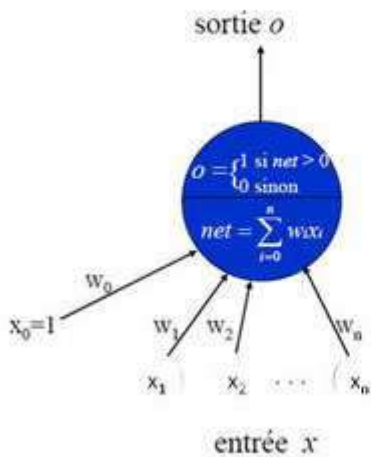
Couche de sortie

- Restitution du résultat
Description de la réponse au problème traité
Transmission à l'utilisateur

Hervé Barbot, 2009-2011

26

Le Perceptron (un réseau acyclique à une seule couche)



Fonction à seuil

Somme pondérée des entrées

Hervé Barbot, 2009-2011

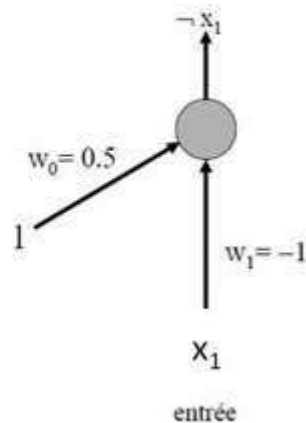
27

Exemples

- Négation logique

entrée x1	sortie
0	1
1	0

Sortie = si $-x_1 + 0,5 > 0$ alors 1
sinon 0



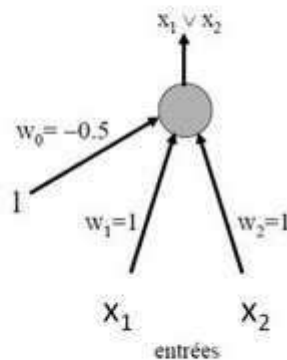
Hervé Barbot, 2009-2011

28

- « ou » logique

entrée x1	entrée x2	sortie
0	0	0
0	1	1
1	0	1
1	1	1

Sortie = si $x_1 + x_2 - 0,5 > 0$ alors 1
sinon 0

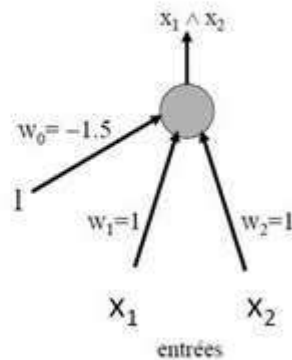


Hervé Barbot, 2009-2011

29

- « et » logique

entrée x1	entrée x2	sortie
0	0	0
0	1	0
1	0	0
1	1	1



Sortie = si $x_1 + x_2 - 1,5 > 0$ alors 1
sinon 0

Hervé Barbot, 2009-2011

30

Problème...

- « xor » logique (le « ou » exclusif)

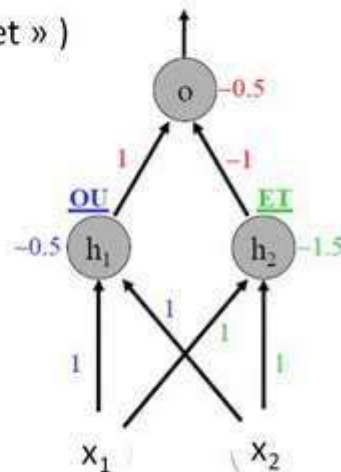
entrée x1	entrée x2	sortie
0	0	0
0	1	1
1	0	1
1	1	0

Hervé Barbot, 2009-2011

31

- « xor » = « ou » et (non « et »)

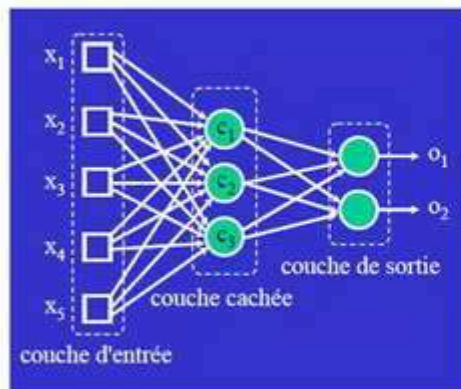
entrée x1	entrée x2	sortie
0	0	0
0	1	1
1	0	1
1	1	0



Hervé Barbot, 2009-2011

32

Réseau acyclique multi-couche



Couche cachée

$$c_j = 1 \quad \text{si} \quad \sum_k w_{jk} x_k > 0$$

$$c_j = 0 \quad \text{sinon}$$

Couche de sortie

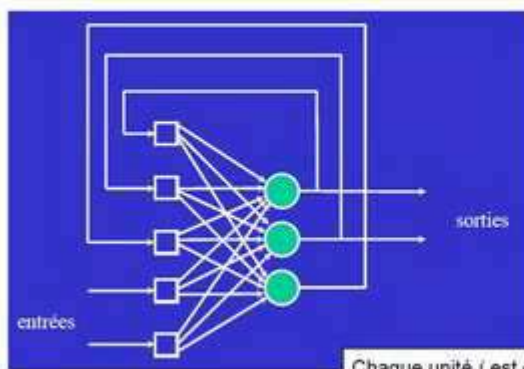
$$o_i = 1 \quad \text{si} \quad \sum_k w_{ik} c_k > 0$$

$$o_i = 0 \quad \text{sinon}$$

Hervé Barbot, 2009-2011

33

Réseau récurrentif (réseau de Hopfield)

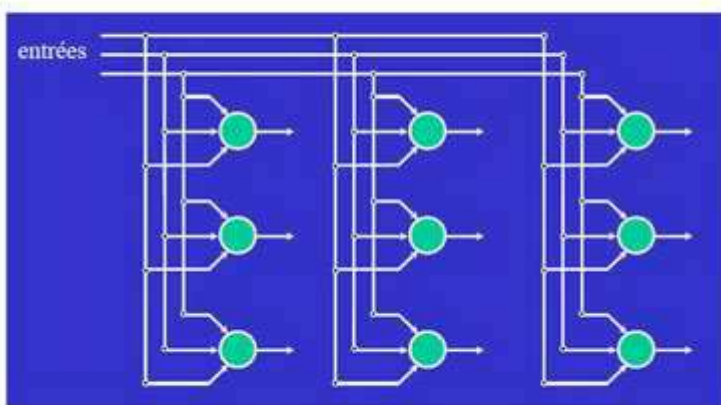


Chaque unité i est connectée à chaque autre unité j par un poids w_{ij}
les poids sont supposés symétriques: $w_{ij} = w_{ji}$

Hervé Barbot, 2009-2011

34

Réseau « en treillis »



Réseau en treillis 3x3 bi-dimensionnel

Hervé Barbot, 2009-2011

35

PROPAGATION DE L'ACTIVATION

DES TRAITEMENTS FAITS DANS LE RÉSEAU

Hervé Barbot, 2009-2011

36

- RNA =
 - un ensemble de neurones
 - interconnectés
 - coopérant
 - en compétition ?
- Flots :
 - de traitement
 - de données
- Organisation des « flots » ?
i.e. « Propagation de l'activation (des neurones) » ?

Hervé Barbot, 2009-2011

37

Propagation de l'activation

- La propagation au niveau du réseau en décrit le comportement collectif
- Cycle d'activation = activation de tous les neurones du réseau
- L'ordre de choix des neurones détermine le traitement, donc les résultats

Hervé Barbot, 2009-2011

38

Propagation de l'activation

- **Synchrone** ----->
 - Asynchrone
 - Séquentielle
 - Aléatoire
- Tous les neurones sont mis à jour en même temps.
 - Chaque neurone calcule son activation au temps t à partir de l'activation que les autres neurones affiche à t-1.

Hervé Barbot, 2009-2011

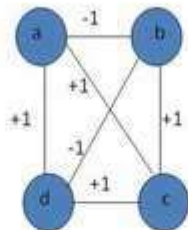
39

Illustration de convergence (Amit 89)

$$a_i = \Gamma(\sum_j w_{ij} * a_j)$$

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

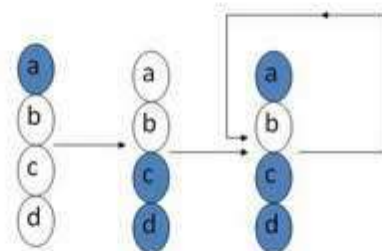
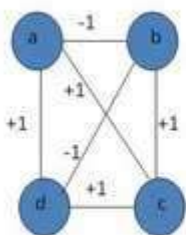
$\Gamma(x)$ Fonction Seuil



Etat du réseau	Etat des neurones			
	a	b	c	d
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	1	0	0
...
16	1	1	1	1

Hervé Barbot, 2009-2011

40




Cycle de Propagation d'activation Synchrone

Hervé Barbot, 2009-2011

41



Propagation de l'activation

- Synchrones
 - **Asynchrone** 
 - Séquentielle
 - Aléatoire
- Les neurones sont mis à jour un à un.
 - Chaque neurone calcule son activation à partir d'activations qui peuvent ou non être recalculées.
 - L'ordre de mise à jour des neurones peut avoir un effet important sur le comportement du réseau.

Hervé Barbot, 2009-2011

42

Propagation de l'activation

- Synchrones
 - **Asynchrone**
 - Séquentielle 
 - Aléatoire 
- La mise à jour s'effectue toujours selon un ordre fixe.
 - l'ordre de mise à jour est aléatoire et est choisi à chaque cycle de propagation

Hervé Barbot, 2009-2011

43

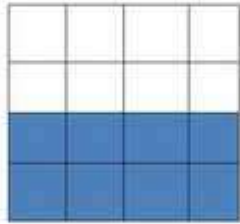
EXEMPLES SIMPLES

Hervé Barbot, 2009-2011

44

Détection de traits

- Ce mécanisme permet au niveau de la rétine de détecter les frontières dans ce qui est perçu.



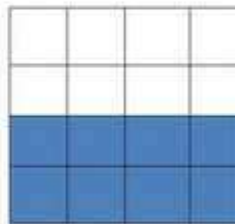
- Un neurone par cellule, seuls les neurones de la frontière seront activés après propagation

Hervé Barbot, 2009-2011

45

- Les neurones de la zone illuminée reçoivent bien une entrée mais elle est étouffée par l'inhibition provenant des autres neurones illuminés.

- Les neurones à la frontière des deux régions ne reçoivent que 1/2 de l'inhibition des autres neurones et demeurent donc actifs



- Les neurones de la zones sombres ne reçoivent aucune entrée et restent donc inactifs.

Hervé Barbot, 2009-2011

46

ERREUR ET APPRENTISSAGE

Hervé Barbot, 2009-2011

47

Introduction

- Le comportement du réseau et ses capacités de calcul dépendent fortement
 - de sa connectivité et
 - des propriétés de propagation.
- Un réglage plus fin est possible en agissant au niveau des poids synaptiques et des connexions
- Le choix d'un ensemble de poids synaptiques adéquats est difficile, d'où
 - l'utilisation de technique d'apprentissage et
 - la mise au point d'un ensemble de règles de contrôle d'apprentissage

Hervé Barbot, 2009-2011

48

Procédure d'apprentissage

- L'apprentissage du réseau se fait dans le contexte d'une tâche, d'un comportement à apprendre
- Les données en entrée sont codées sous forme d'un vecteur couramment appelé *patron d'entrée*.
- L'apprentissage se fait généralement sur une période relativement longue où les patrons d'entrée sont présentés au réseau plusieurs fois

Hervé Barbot, 2009-2011

49

- L'apprentissage comprend essentiellement quatre phases:
 - *Initialisation des poids synaptiques*: Les poids trouvés par le réseau à la fin de la phase d'apprentissage dépendent en partie des valeurs initiales, généralement faibles valeurs aléatoires au début.
 - *Présentation du patron d'entrée et propagation d'activation*
 - *Calcul de l'erreur sur le résultat obtenu*
 - *Calcul du vecteur de correction pour mise à jour des poids synaptiques*

Hervé Barbot, 2009-2011

50

Apprentissage du perceptron

Entrées: ensemble d'apprentissage $\{(x_1, x_2, \dots, x_n, t)\}$

Méthode

initialiser aléatoirement les poids $w(i)$, $0 \leq i \leq n$

répéter jusqu'à convergence:

pour chaque exemple

calculer la valeur de sortie o du réseau.

ajuster les poids:

$$\Delta w_i = \eta (t - o) x_i \quad \text{Règle d'apprentissage du Perceptron}$$

$$w_i \leftarrow w_i + \Delta w_i$$

Hervé Barbot, 2009-2011

51

Apprentissage du perceptron

Entrées: ensemble d'apprentissage $\{(x_1, x_2, \dots, x_n, t)\}$

Méthode

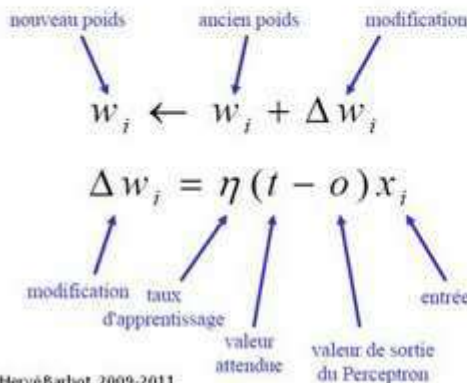
initialiser aléatoirement les poids $w(i)$, $0 \leq i \leq n$

répéter jusqu'à convergence:

pour chaque exemple

calculer la valeur de sortie o du réseau.

ajuster les poids:



$$\Delta w_i = \eta (t - o) x_i \quad \text{Règle d'apprentissage du Perceptron}$$

$$w_i \leftarrow w_i + \Delta w_i$$

Hervé Barbot, 2009-2011

52

Apprentissage par Rétro-Propagation (Rumelhart, Hinton et Williams)

- Phase en avant
 - Construction de l'ensemble entraînement, constitué d'une paire de valeurs (valeur d'entrée et valeur de sortie désirée) et l'introduction au réseau de cet ensemble pour le calcul de la valeur de sortie du réseau.
- Phase en arrière
 - Calcul de l'erreur (écart) entre *la valeur actuelle* de la *sortie* et celle de la *sortie désirée* et l'ajustement des poids afin de minimiser l'erreur.
- Répétition pour toutes les paires de l'ensemble entraînement jusqu'à minimisation de l'erreur.

Hervé Barbot, 2009-2011

53

- Initialiser les poids à de petites valeurs aléatoires (entre -0.05 et 0.05)
- Tant que la condition d'arrêt n'est pas atteinte
 - Pour chaque exemple d'entraînement

- Calculer les sorties du réseau.
- Pour toutes les unités de sortie k , calculer l'erreur:

$$\delta_k \leftarrow o_k(1 - o_k)(t_k - o_k)$$

- Pour toutes les unités cachées h , calculer l'erreur:

$$\delta_h \leftarrow o_h(1 - o_h) \sum_{k \in \text{sorties}} w_{kh} \delta_k$$

- Mettre à jour tous les poids w_{ji} :

$$w_{ji} \leftarrow w_{ji} + \Delta w_{ji}$$

$$\Delta w_{ji} = \eta \delta_j x_{ji}$$

x_{ji} : l'entrée qui provient de l'unité i vers l'unité j
 w_{ji} : le poids correspondant

10/25

© Sébastien Paquet (2004)

22

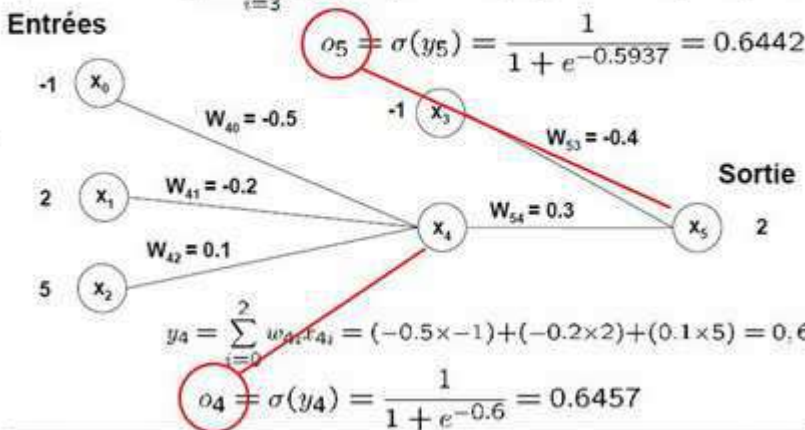
Hervé Barbot, 2009-2011

54

Exemple

$$\eta = 0.05 \quad y_5 = \sum_{i=3}^4 w_{5i} x_{5i} = (-0.4 \times -1) + (0.3 \times 0.6457) = 0,5937$$

$$o_5 = \sigma(y_5) = \frac{1}{1 + e^{-0.5937}} = 0.6442$$



10/25

© Sébastien Paquet (2004)

24

Hervé Barbot, 2009-2011

55

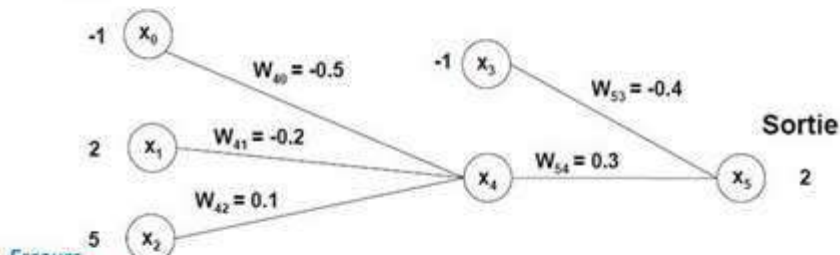
Exemple

$$\eta = 0.05$$

$$o_4 = 0.6457$$

$$o_5 = 0.6442$$

Entrées



Erreurs

$$\delta_5 = o_5(1 - o_5)(t_5 - o_5) = 0.6442(1 - 0.6442)(2 - 0.6442) = 0.3108$$

$$\delta_4 = o_4(1 - o_4) \sum_{k \in \{5\}} w_{k4} \delta_k = 0.6457(1 - 0.6457)(0.3 \times 0.3108) = 0.0213$$

10/25

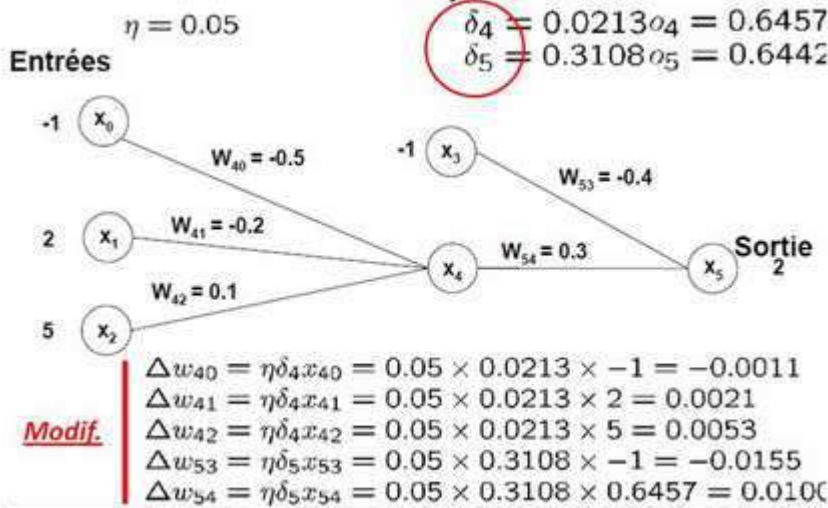
© Sébastien Paquet (2004)

25

Hervé Barbot, 2009-2011

56

Exemple



9.7.7a

© Sébastien Paquet (2004)

26

Hervé Barbot, 2009-2011

57

Condition d'arrêt

- Le nombre d'itérations est important car:
 - Si trop faible, l'erreur n'est pas suffisamment réduite.
 - Si trop grand, le réseau devient trop spécifique aux données d'entraînement.
- Il y a plusieurs conditions d'arrêt possible:
 - Après un certain nombre fixe d'itérations.
 - Lorsque l'erreur dans les sorties des exemples d'entraînement descend en dessous d'une certaine borne.
 - Lorsque l'erreur avec les exemples de validation descend en dessous d'une certaine borne.

9.7.7a

© Sébastien Paquet (2004)

28

Hervé Barbot, 2009-2011

58

Types d'apprentissage

- On distingue trois groupes d'apprentissages:
 - **Apprentissage non supervisé** : permet de modifier les poids du réseau en fonction d'un critère interne.
 - **Apprentissage supervisé** : dispose d'un comportement de référence vers lequel on tente de faire converger le réseau
 - **Apprentissage semi-supervisé** : suppose qu'il est possible d'avoir des indications qualitatives sur les performances du réseau

Hervé Barbot, 2009-2011

59