APPRENTISSAGE ET GENERALISATION PAR DES RESEAUX DE NEURONES : ETUDE DE NOUVEAUX ALGORITHMES CONSTRUCTIFS

Juan Manuel Torres Moreno

Septembre 22, 1997

CEA/Grenoble
Département de Recherche Fondamentale sur la Matière Condensée/SPSMS/Groupe Théorie
17 rue des Martyrs 38054 Grenoble Cedex 9

PLAN

1. INTRODUCTION

   CLASSIFICATION ET RESEAUX DE NEURONES

2. L'ALGORITHME D'APPRENTISSAGE MINIMERROR

3. DEUX ALGORITHMES CONSTRUCTIFS

   MONOPLAN ET NETLINES

4. CONCLUSION ET PERSPECTIVES

LA CLASSIFICATION

Classification. Assignation d'une classe à un objet à partir de ses propriétés spécifiques.

Exemples

- **Classification des visages**
- **des données médicales**
- **de caractères manuscrits**
- **de spectres...**

Apprentissage. Processus d'adaptation des paramètres d'un système pour donner une réponse désirée face à une entrée ou stimulation externe.

Données. Paires { Entrées-Sortie } sous une forme vectoriel.

$$\left.\begin{array}{l}\vec{\xi}=\left(\xi_1,\xi_2,\ldots,\xi_N\right)\\ \tau=\pm 1\end{array}\right\}$$ **= EXEMPLE**

Ensemble d'apprentissage. P couples de N entrées (binaires ou réelles)

$$\mathcal{L}=\left\{\vec{\xi}^{\mu},\tau^{\mu}\right\};\quad \mu=1,\ldots,P$$

LA GENERALISATION

But de l'apprentissage...

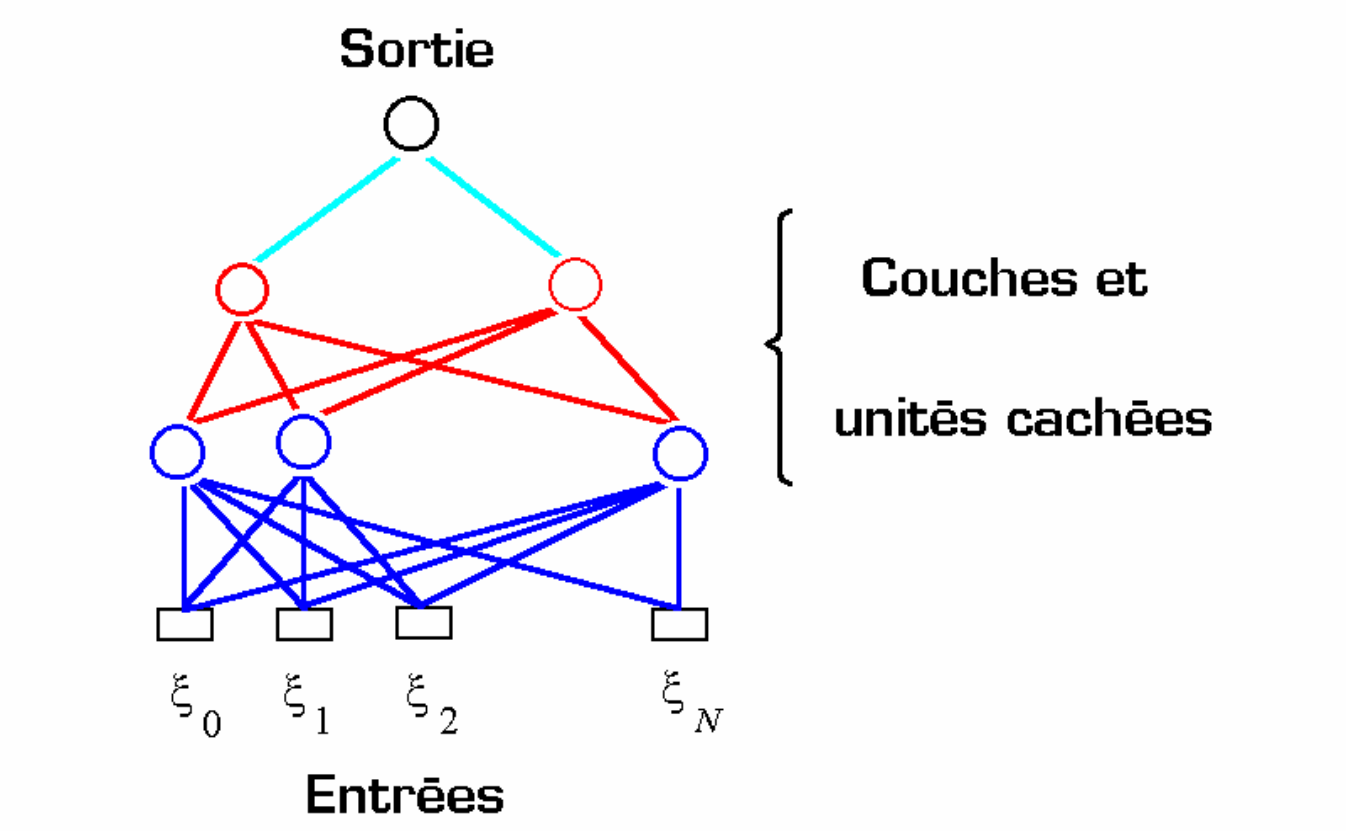
Classer de nouveaux exemples (dont on ne connaît pas la classe) qui n'appartienent pas a l'ensemble d'apprentissage.

Mesure de l'erreur de généralisation

$$\varepsilon_g = \frac{\text{Nb. total de bien classés}}{\text{Nb. total d'exemples}}$$

RESEAUX DE NEURONES

Ensemble d'unités (neurones) inter-connectées par des POIDS, qui traitent l'information d'une source externe.



Réseau *feedforward* à deux couches cachées

Un RN avec une seule couche cachée peut approcher toute fonction des entrées, mais le nombre d'unités cachées nécessaires est inconnu...

Deux méthodes

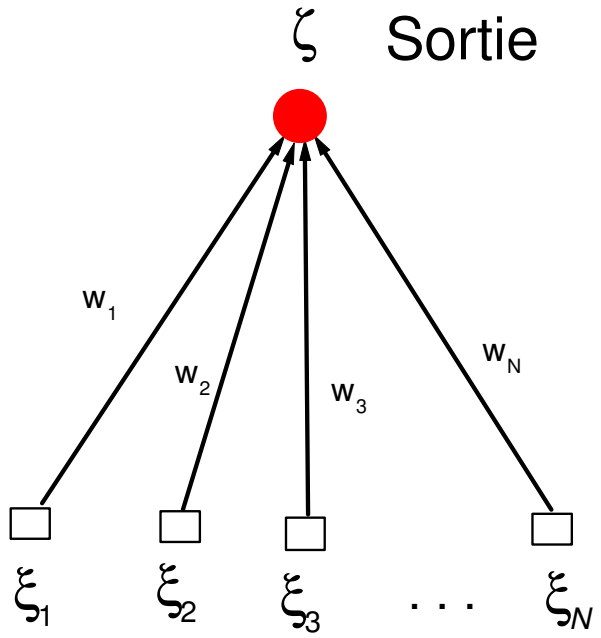
Architecture fixe : Backprop et variations

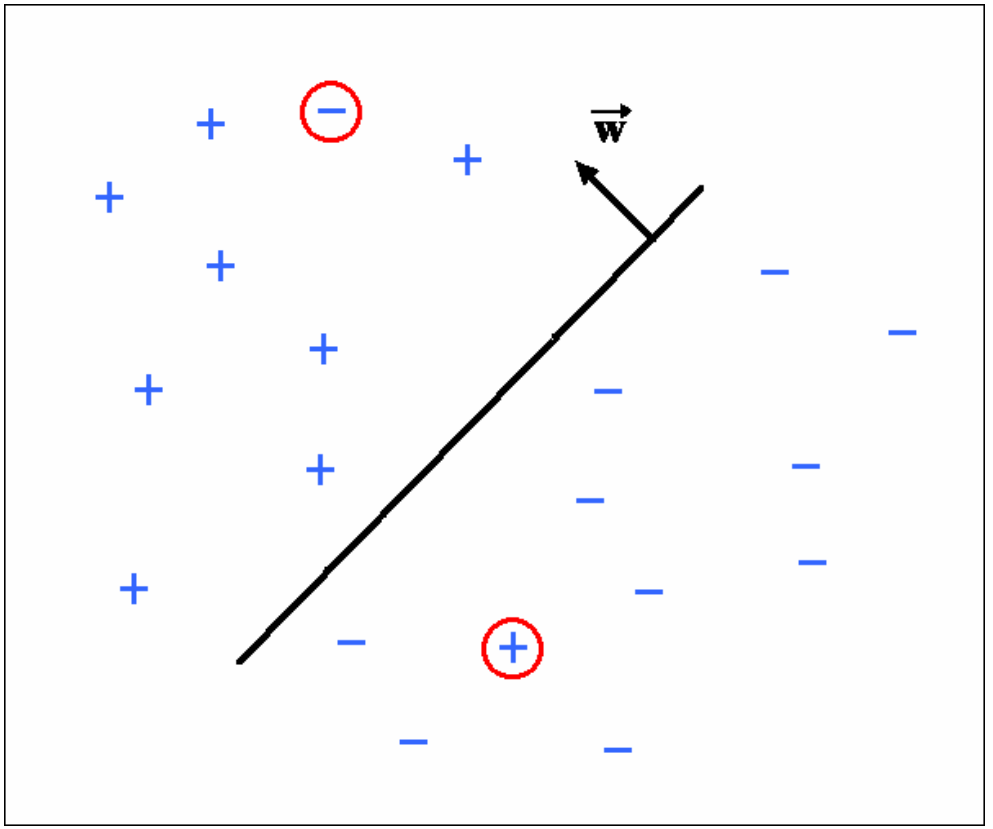
- Nombre de couches et d'unités fixés : apprentissage des poids.
- Détermination de la taille du réseau par *essai et erreur*. (Élagage éventuel).

Architecture constructive : Algorithmes incrémentaux

- Nombre d'unités et poids déterminés par apprentissage.
- Commencer avec une unité, et introduire successivement des neurones cachés.

LE PERCEPTRON





- **Les *N* entrées** ξ_i **;** **i=1,..., *N***
- **Les *N* poids** w_i **; i=1,..., *N***
- **La sortie**

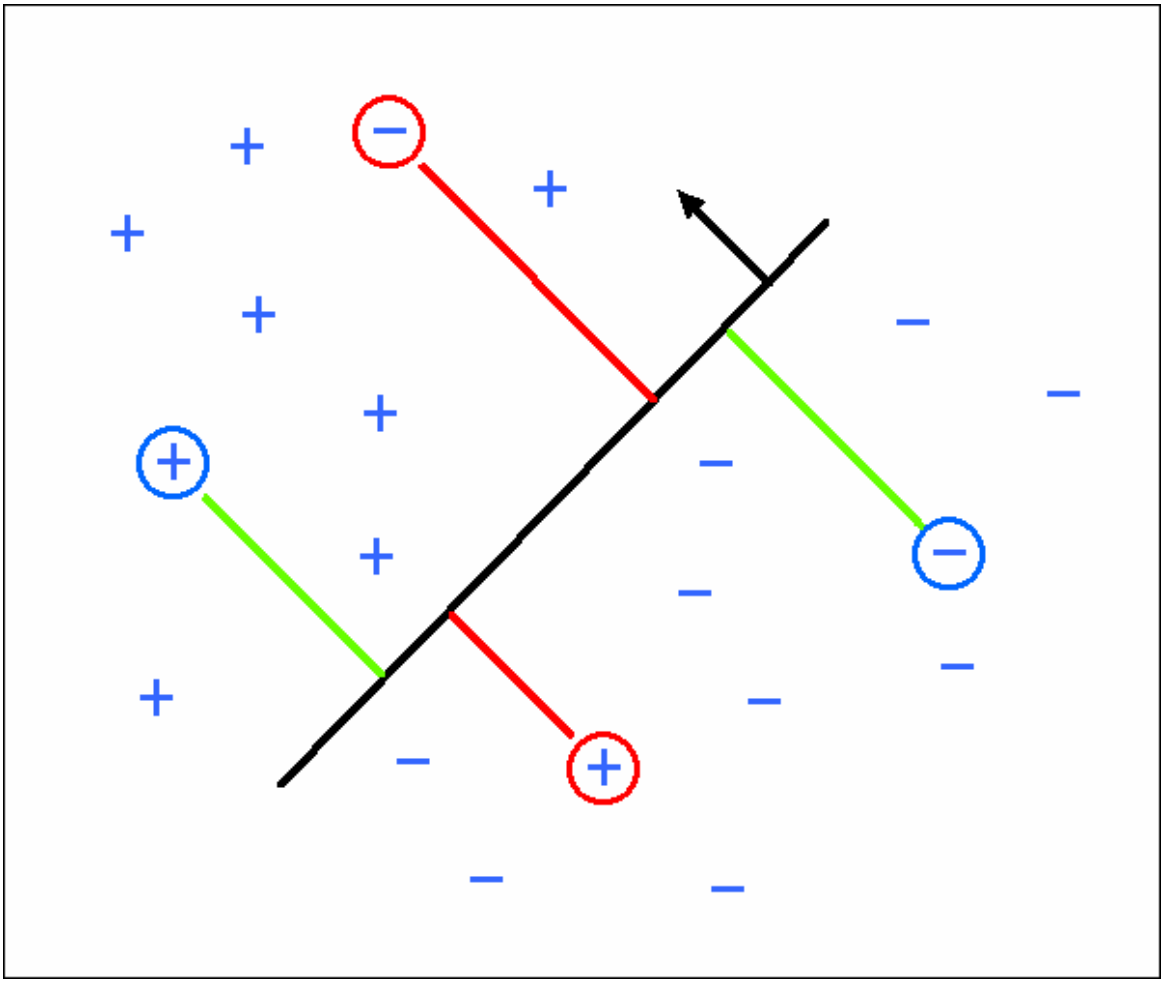
$$\vec{\xi} = (\xi_0, \xi_1, ..., \xi_N)$$

$$\vec{w} = (w_0, w_1, ..., w_N)$$

$$\zeta = \text{signe}(\vec{w} \cdot \vec{\xi})$$

$\xi_0 \equiv 1$ = BIAIS

LA STABILITE



$$\gamma^{\mu} \equiv \frac{\vec{\mathbf{w}}}{\|\vec{\mathbf{w}}\|} . \vec{\xi}^{\mu} \, \tau^{\mu} \qquad \gamma^{\mu} = \begin{cases} > 0 \text{ bien classé} \\ < 0 \; mal \text{ classé} \end{cases}$$

Distance (avec signe) de l'exemple μ à l'hyperplan séparateur.

Une grande stabilité positive assure une certaine robustesse de la réponse du neurone.

FONCTION DE COUT

- **Mesure d'erreur.**

$$E(\vec{w}) = \sum_{\mu=1}^{P} \Theta(-\gamma^{\mu}) \qquad \Theta(x) = \begin{cases} 1 \text{ si } x > 0 \\ 0 \text{ autrement} \end{cases}$$

Mais Θ n'est pas ***dérivable...***

Pour des unités binaires, il existe algorithmes d'apprentissage :

- **L'algorithme standard du perceptron**
- **L'algorithme *Pocket***

Problèmes

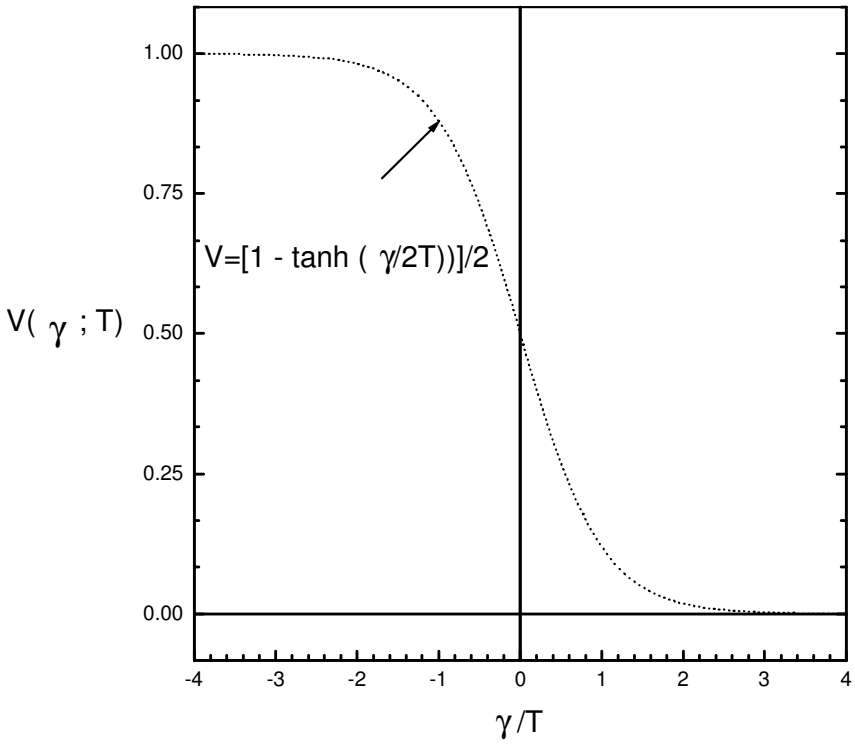
- **L'algorithme standard converge seulement si l'ensemble $\mathcal{L}$ est linéairement séparable**
- **Pour l'algorithme *Pocket*, il faut atteindre un temps *suffisamment long*...**

L'ALGORITHME MINIMERROR

Minimise la fonction de coût :

$$E(\vec{w}) = \sum_{\mu=1}^{P} V(\gamma^{\mu}) \; ; \qquad V(\gamma^{\mu}) = \frac{1}{2}\left[1 - \tanh \frac{\gamma^{\mu}}{2T} \right]$$

T = Température ***γ= Stabilité***



Rôle de T :

$T \to 0 \quad \Rightarrow \quad V$ échelon : compte les erreurs

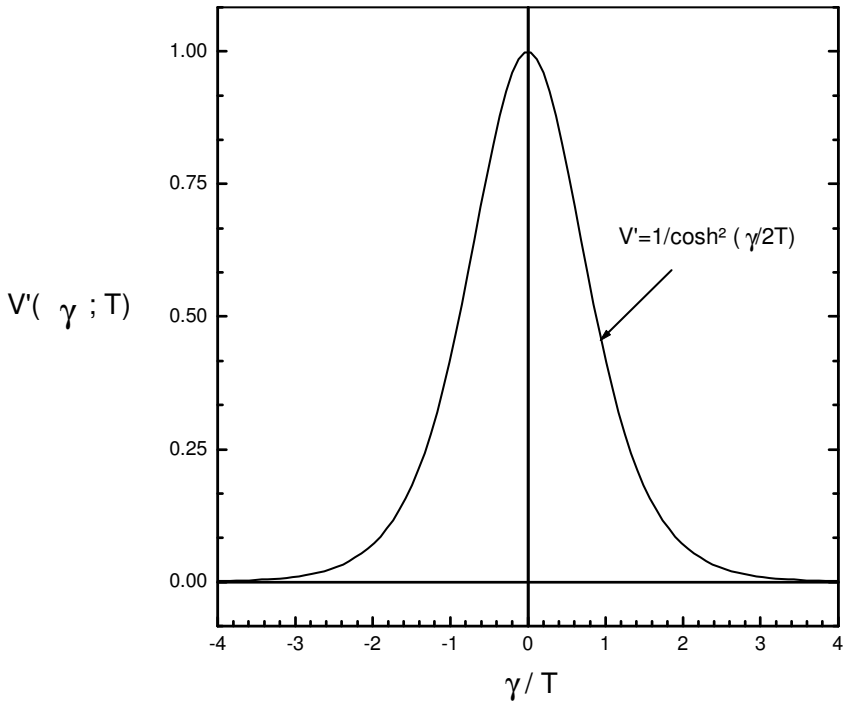
T finie $\quad \Rightarrow \quad V$ est *dérivable*...

A T finie on peut utiliser une descente en gradient...

$$\vec{\mathbf{w}}(t+1) = \vec{\mathbf{w}}(t) + \delta\vec{\mathbf{w}}(t)$$

$$\delta\vec{\mathbf{w}}(t) = -\varepsilon \frac{\partial E}{\partial \vec{\mathbf{w}}}$$

$$\frac{\partial E}{\partial \vec{\mathbf{w}}} = \sum_{\mu} \frac{\partial V}{\partial \vec{\mathbf{w}}} = -\frac{1}{4T} \sum_{\mu} \frac{\vec{\xi}^{\mu}}{\cosh^2\left(\frac{\gamma^{\mu}}{2T}\right)} \tau^{\mu}$$



- **Propriété : utiliser l'information des exemples *dans une fenêtre* de largeur $2T$**

Minimisation de la fonction de coût :

$$C = \frac{1}{2} \sum_{\mu=1}^{P} \left[1 - \tanh\left(\gamma^{\mu} / 2T \right) \right]$$

$\gamma^{\mu} = \tau^{\mu} \vec{\mathbf{w}} \cdot \vec{\xi}^{\mu} / \left\| \vec{\mathbf{w}} \right\|$: stabilité des entrées

($\gamma^{\mu} > 0$ exemple bien appris, $\gamma^{\mu} \leq 0$ autrement)

Minimisation : **gradient simple + recuit déterministe**

Propriétés

Trouver automatiquement un perceptron qui :

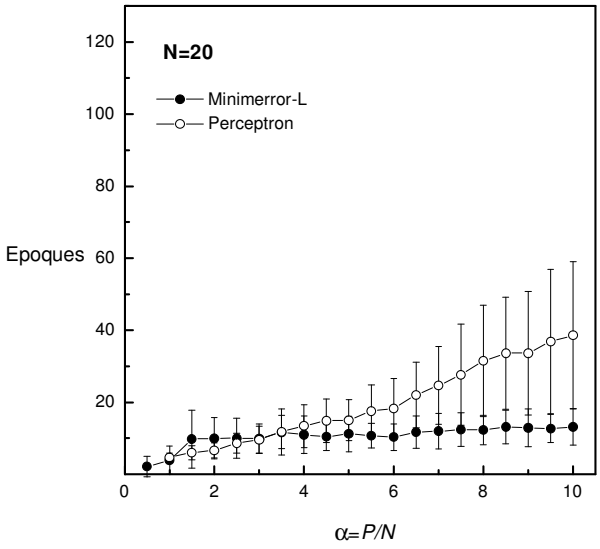
- Si $\mathcal{L} = \{ \vec{\xi}^{\mu}, \tau^{\mu} \}$ est LS $\Rightarrow$ **probabilité de généralisation maximale**
- Si non $\Rightarrow$ probabilité d'**erreur d'apprentissage minimale**

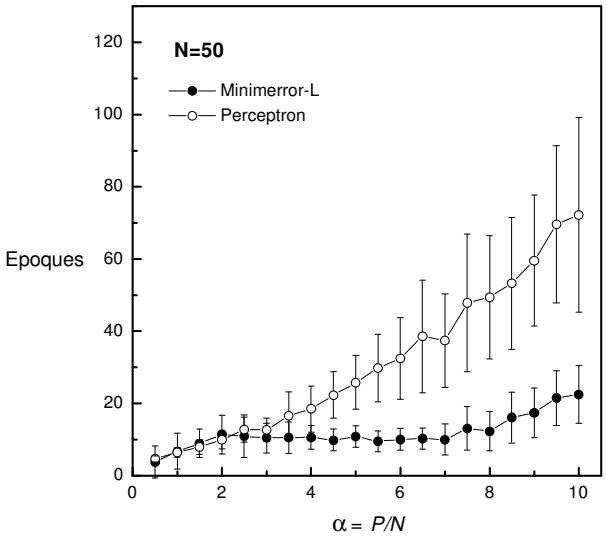
Minimerror vs. algorithme standard du Perceptron

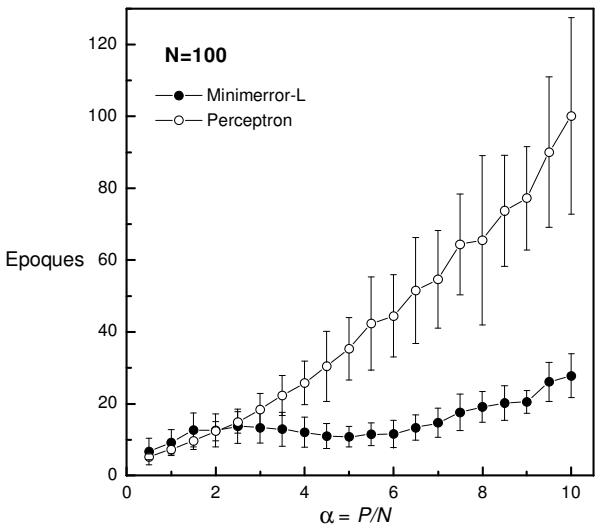
$$\alpha = \frac{P}{N}$$

Taille réduite de l'ensemble d'apprentissage

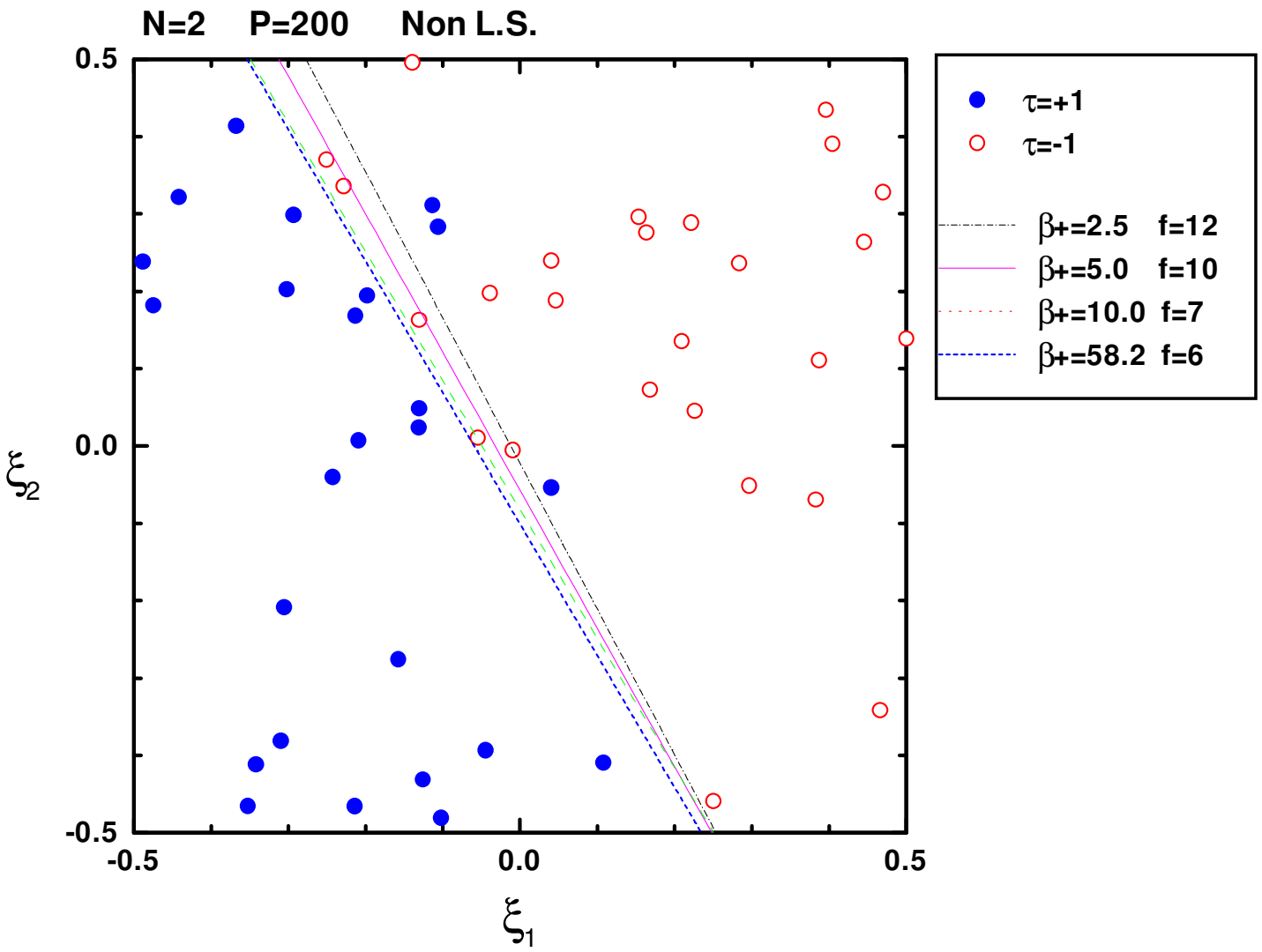
Moyennes sur 30 ensembles d'apprentissage





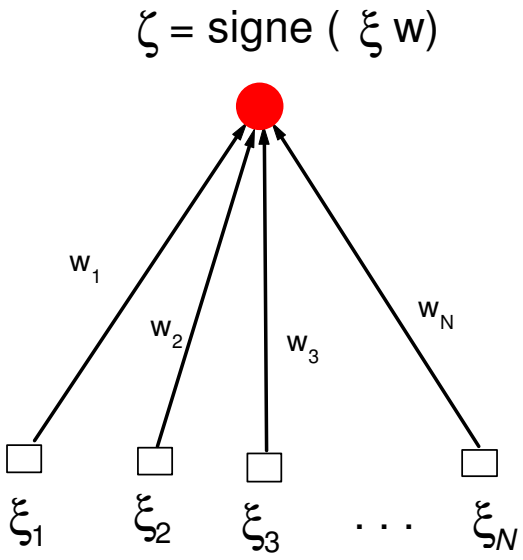


INFLUENCE DE T ($\beta=1/T$)

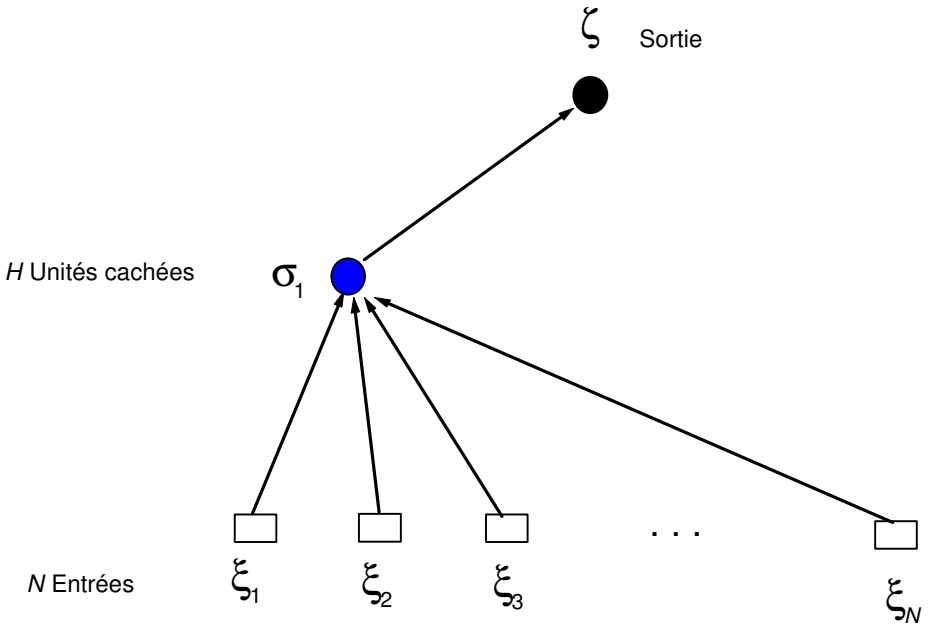


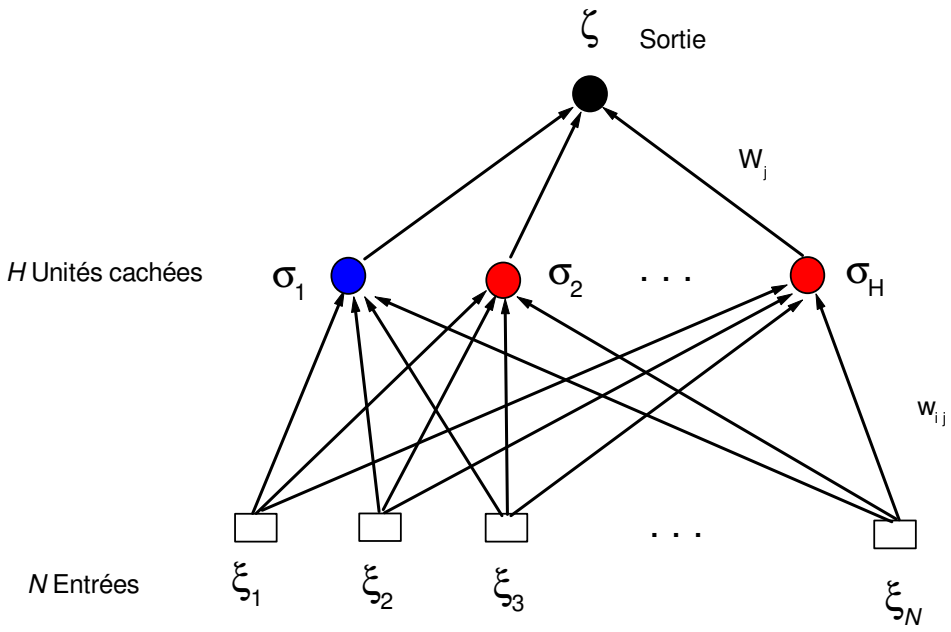
ALGORITHMES CONSTRUCTIFS

A partir de perceptrons simples...



Bâtir un réseau de neurones...




ζ
Sortie
W_j
H Unités cachées
σ_1
σ_2
. . .
σ_H
w_ij
. . .
N Entrées
ξ_1
ξ_2
ξ_3
ξ_N

L'ALGORITHME CONSTRUCTIF MONOPLAN*

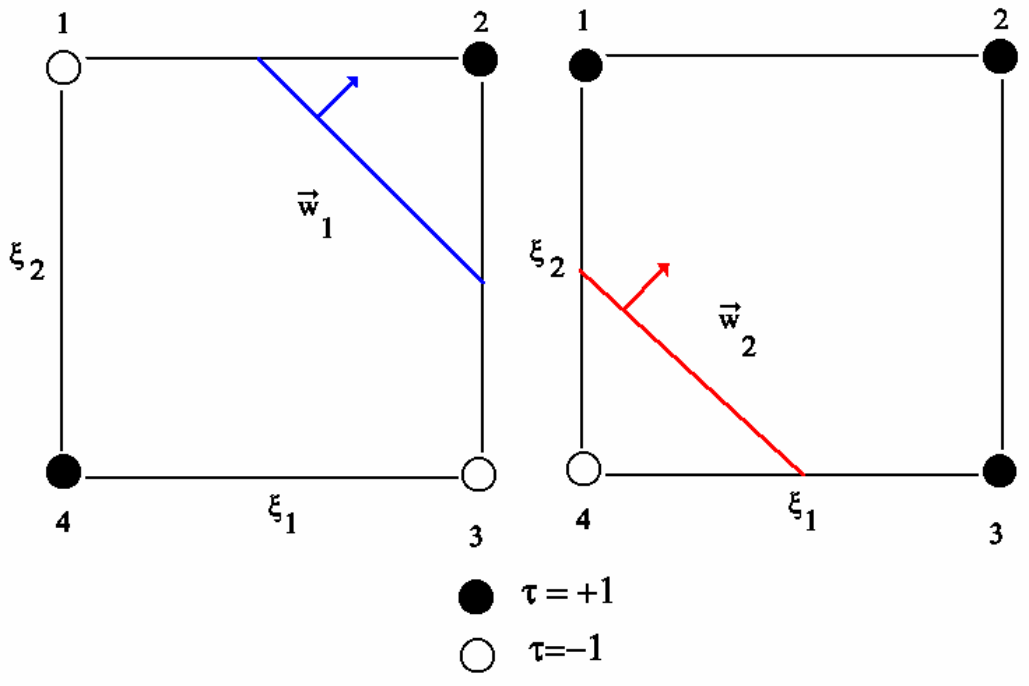
Il construit un RN *feedforward* **...**

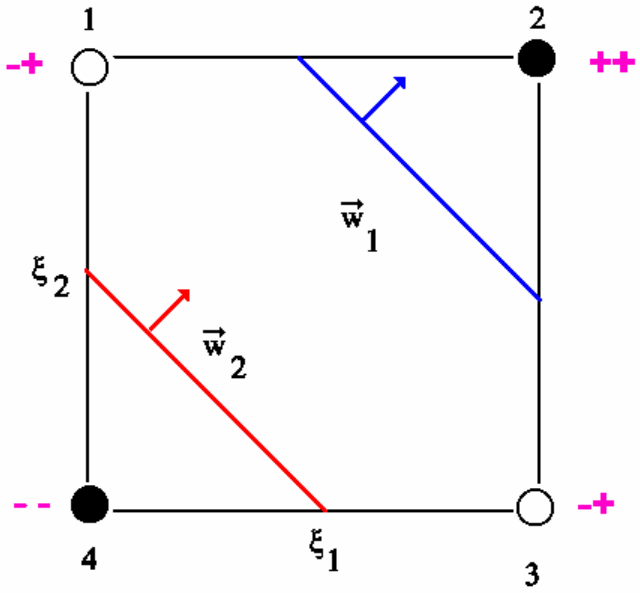
- Réseau à une seule couche cachée de *neurones binaires*
- Apprentissage par des perceptrons simples avec **MINIMERROR**
- Les états appris constituent les représentations internes (**RI**) $\vec{\sigma}^{\mu}$ des entrées $\vec{\xi}^{\mu}$

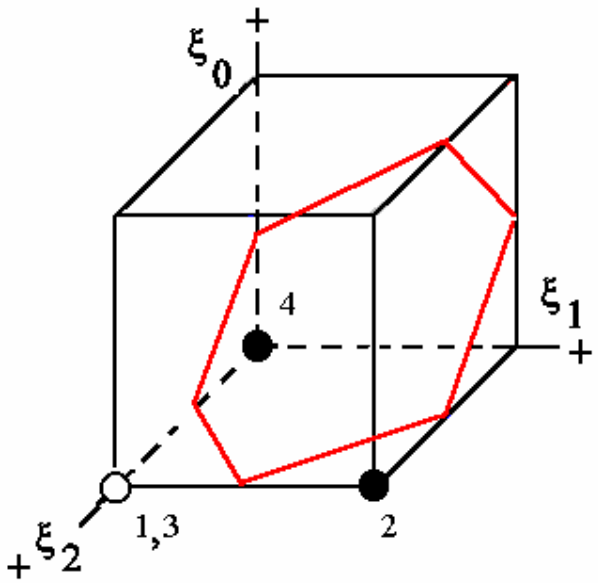
- **Les RI sont un *codage comprimé* (binaire) qui permet d'extraire de règles.**
- **Les poids w sont la *définition de frontières* (ou morceaux de frontières) entre classes.**

♣ **Torres-Moreno et Gordon, ESANN'95 pp 365-370**

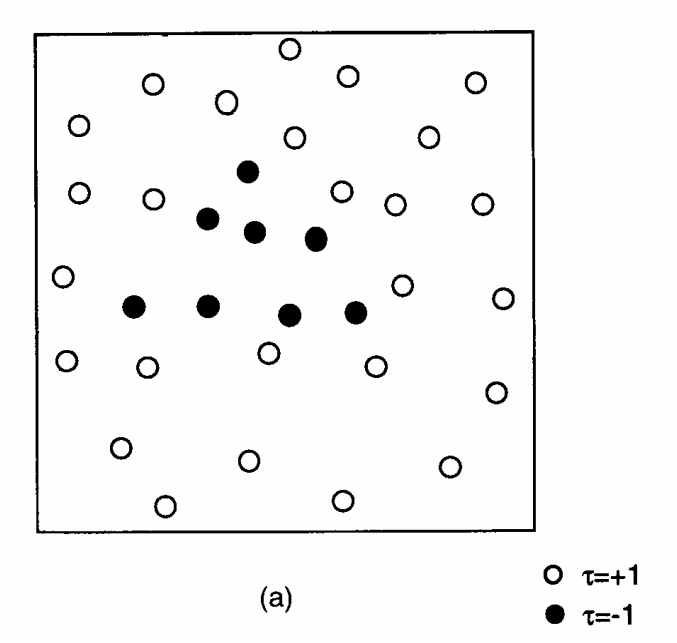
MONOPLAN ET LE XOR



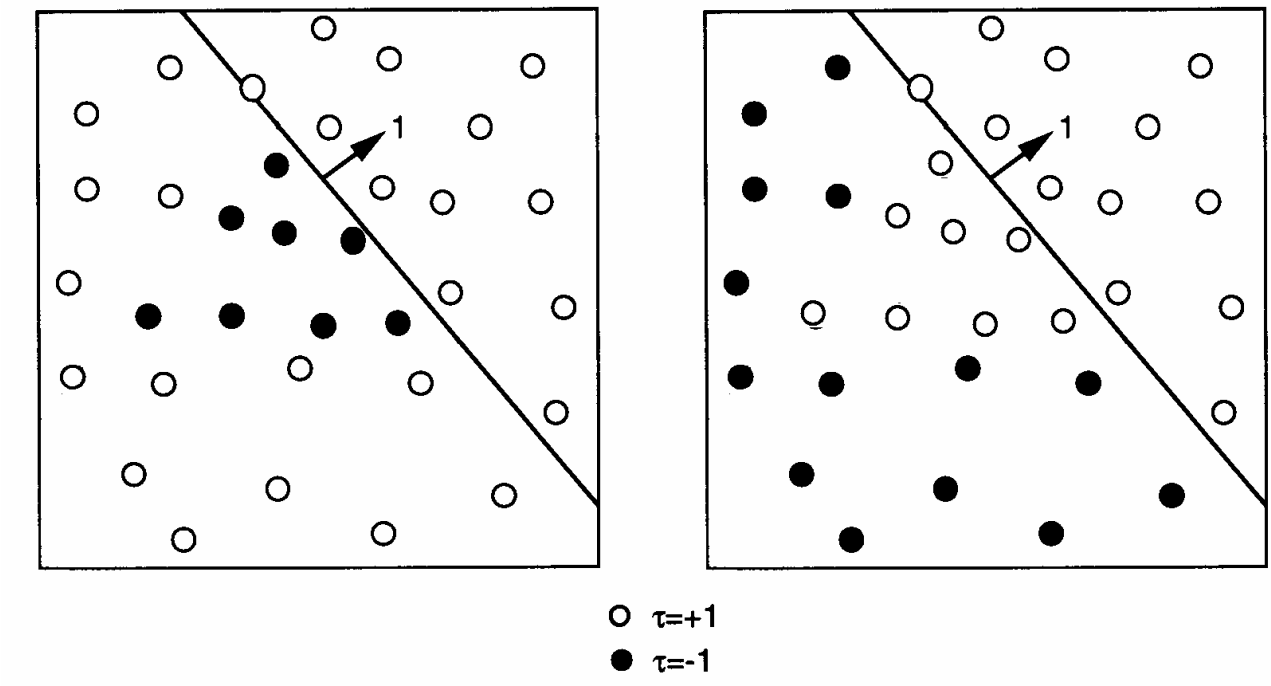


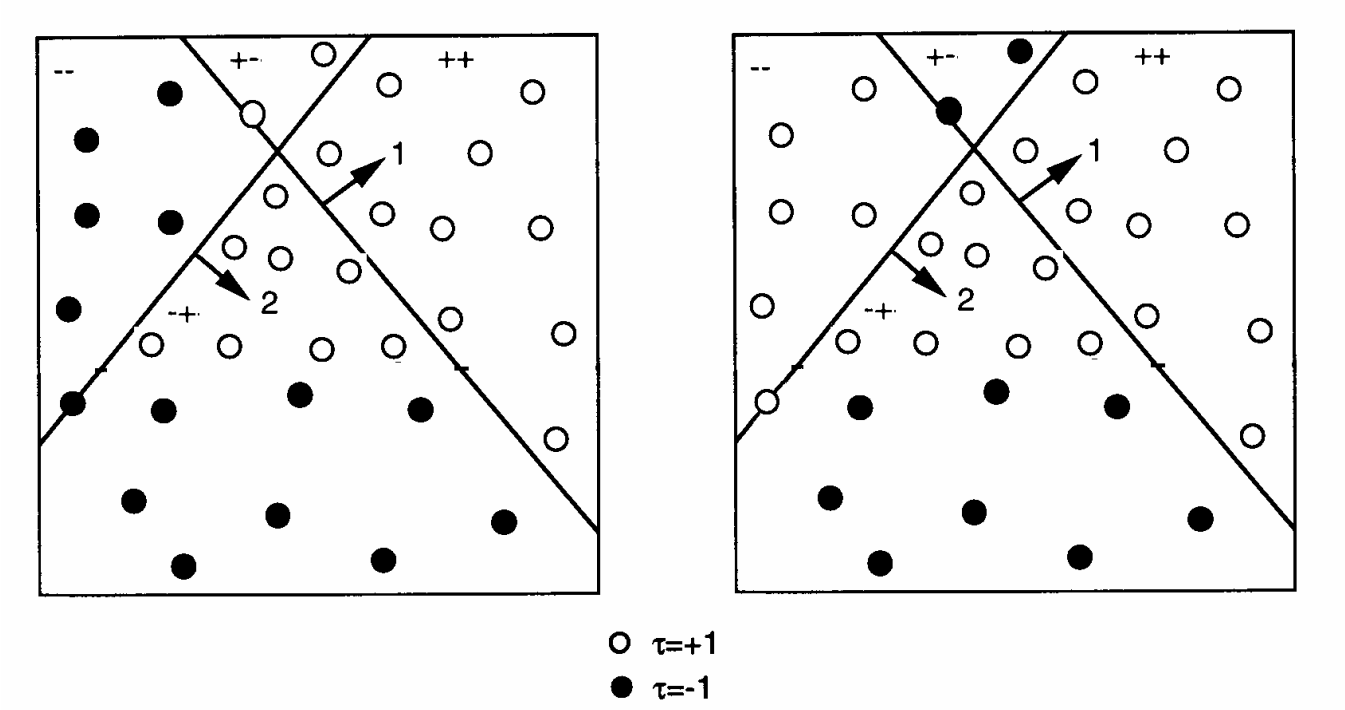


Problème : Malgré sa simplicité, Monoplan peut avoir certains problèmes quand les entrées sont réelles...

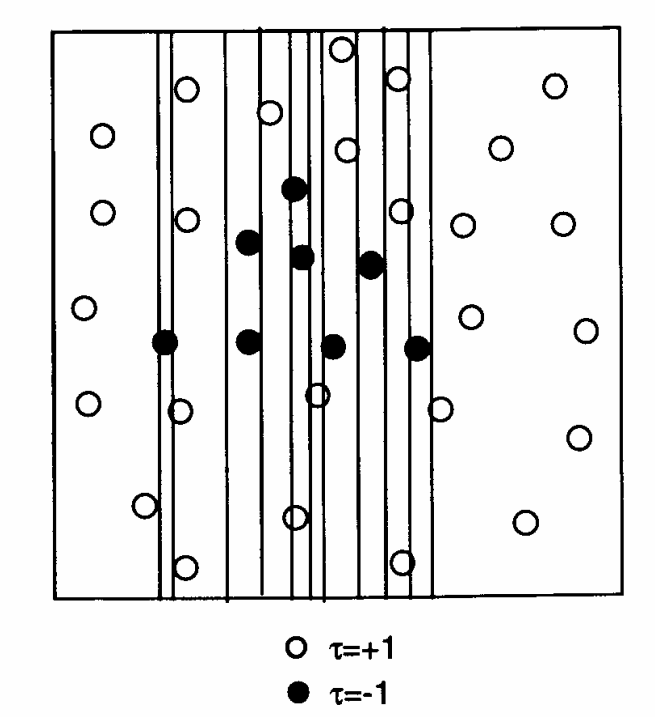


Solution de Monoplan...





Solution compatible avec le théorème de convergence
(Martinez et Stève 1992 ; Gordon 1996)

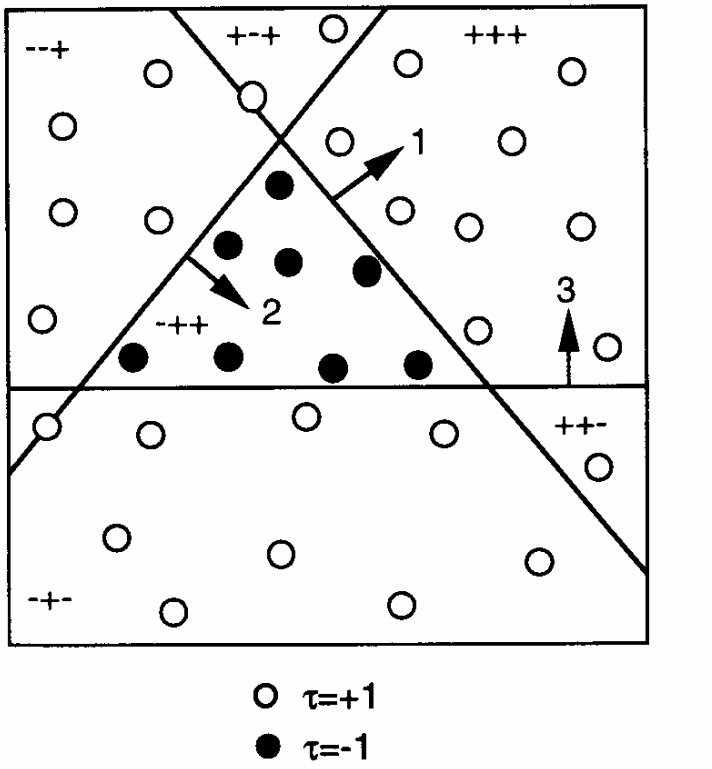


Problèmes :

Trop d'unités cachées... Mauvaise généralisation.

L'ALGORITHME NETLINES

Idée : corriger les erreurs à la sortie et non dans la couche cachée...



NetLines trouve des RI fidèles : deux entrées de classes différentes ont des RI différentes...
et linéairement séparables !

Problèmes artificiels

- La parité de N entrées.
- Les problèmes de Monk's.
- Les formes d'ondes (Breiman).

Classification de spectres

- Classification de données de sonar (Sejnowski).

Aide au diagnostic médical

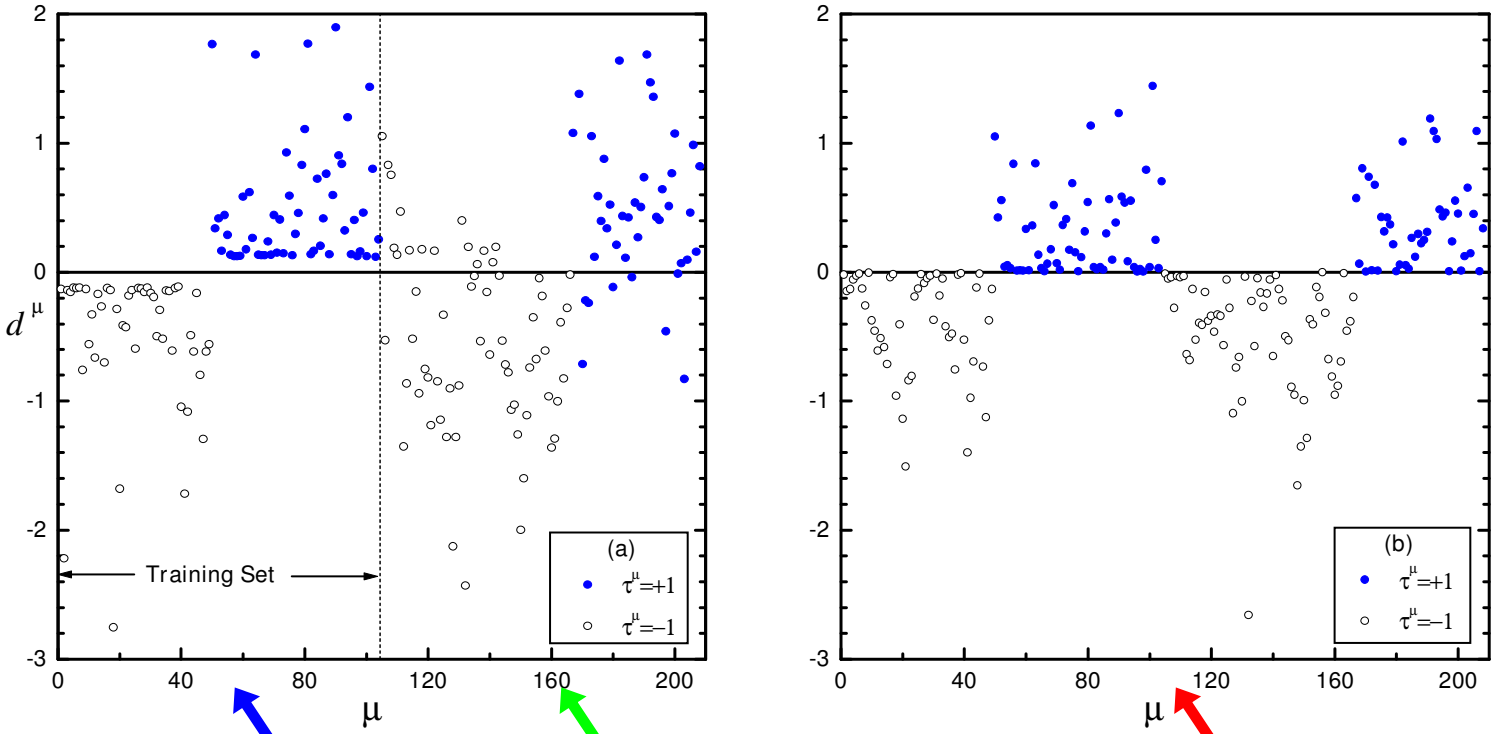
- Diagnostic de cancer du sein (Wisconsin university).
- Diagnostic du diabètes (indiens Pima).

Autres problèmes...

- La base de données Iris (Fisher).
- Le problème de 2 domaines ou plus

Le problème du Sonar (Sejnowski)♣

Discriminer entre échos des mines ou de pierres $P=104$ $N=60$



$\varepsilon_t=0$ $\varepsilon_g=0.22$ $\varepsilon_t=0$

♣Neural Processing Letters (1997) à paraître.

Diagnostic de cancer du sein

Problème

Données médicales de 683 prélèvements cytologiques, plus 16 cas avec l'attribut 6 manquant♣.

Attributs

1	Épaisseur de l'échantillon
2	Uniformité de la taille
3	Uniformité de la forme
4	Adhésion marginale
5	Taille cellule épithéliale
6	Noyaux
7	Chromatine terne
8	*Nucleoli* normal
9	Mitose

Normalisation des données :

$$\xi_i^\mu \leftarrow \frac{\left(\xi_i^\mu - \langle x_i \rangle\right)}{\sigma} ; \quad \langle x_i \rangle = \frac{1}{P} \sum_{\mu=1}^{P} \xi_i^\mu ; \quad \sigma^2 = \frac{1}{P} \sum_{\mu=1}^{P} \left(\xi_i^\mu - \langle x_i \rangle\right)^2$$

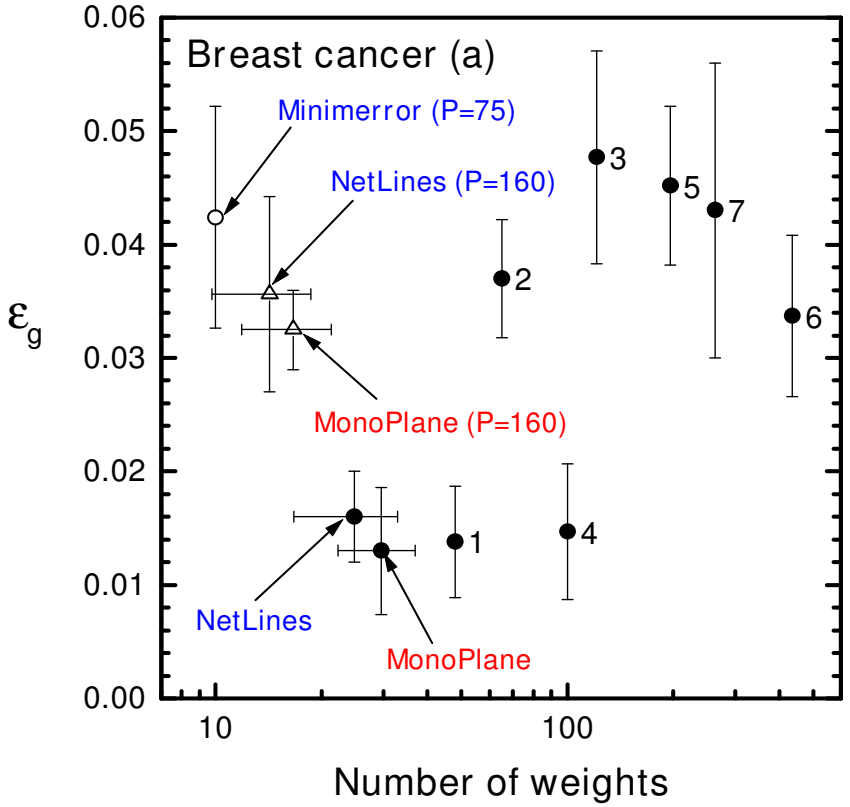
Apprentissage

N	9 attributs $\in$ [1,10]
P	75, 160 et 525
Classes	{bénin, malin}
Distribution	65.5% bénin, 34.5% malin
Ensembles	50 au hasard

Généralisation

Taille de la base	G = 608,523 et 158 respect.
Type de test	*Holdout*

Diagnostic du cancer du sein♣



Retropropagation de l'erreur (sans court circuits)

1 : H=4+2

2 : H=4+4

3 : H=8 + 4

Retropropagation de l'erreur (court circuits)

4 : H=4+2

5 : H=4+4

6 : H=8+4

7 Cascade Correlation

♣Neural Computation (1997) à paraître.

Rprop 10 tests, 2 couches cachées : L. Prechelt: Report 21/94 Fakultät für Informatik, Universität Karlsruhe, Germany (1994)

Glocal, C.Correlation : J. Depenau: Proc. of the World Congress on Neural Networks, Washington. Vol.1, pp. 587-590 (1995).

W. H. Wolberg and O. L. Mangasarian: Proc.of the National Academy of Sciences 87, 9193-9196 (1990).

Diagnostic de diabète

Problème

Données médicales de 768 cas des indiens Pima (*National Institute of Diabetes and Digestive and Kidney Diseases*)♣.

Attributs

1	Nombre de fois prégnant
2	Concentration de glucose 2 heures après d'un test oral de tolérance
3	Pression diastolique de la sang (mmHg)
4	Épaisseur de peau dans triceps (mm)
5	2 heures *sérum* insuline (mu U/ml)
6	Indice de la masse du corps (kg/m^2)
7	Fonction de prédisposition au diabète
8	Âge (années)

Apprentissage

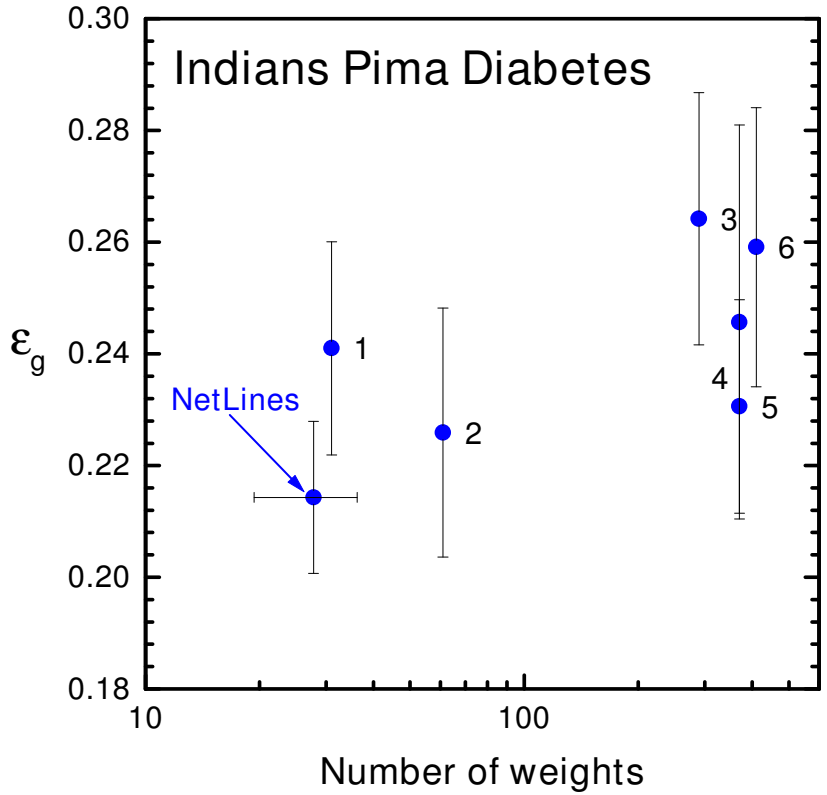
N	8 attributs $\in \Re$
P	576
Classes	{oui, non}
Distribution	65.1% non, 34.9% oui
Ensembles	50 au hasard

Généralisation

Taille de la base	$G = 192$
Type de test	*Holdout*

Normalisation des données: $\xi_i^\mu \leftarrow \frac{(\xi_i^\mu - \langle x_i \rangle)}{\sigma}$; $\langle x_i \rangle = \frac{1}{P}\sum_{\mu=1}^{P} \xi_i^\mu$; $\sigma^2 = \frac{1}{P}\sum_{\mu=1}^{P} (\xi_i^\mu - \langle x_i \rangle)^2$

Diagnostic de diabètes♣



1 Rprop 2+2 No shortcut
2 Rprop 4+4 No shortcut
3 Rprop 16+8 No shortcut
4 Rprop 2+2 shortcut
5 Rprop 4+4 shortcut
6 Rprop 16+8 shortcut

Choix entre 12 architectures differentes:

1 couche cachée: H=2,4,8,16,24,32
2 couches cachées: 2+2,4+2,4+4,8+4,8+8,16+8

♣Neural Computation (1997) à paraître.

Rprop 10 tests: L. Prechelt: Report 21/94 Fakultät für Informatik, Universität Karlsruhe, Germany (1994)

NetLines : 50 **ensembles d'apprentissage**

Données en http://www.ics.uci.edu//~mlearn/MLRepository.html

Formes d'ondes de Breiman♣

Problème

Classement d'ondes $\vec{x}$ appartenant à 3 classes.

Définition

$Classe1$: $\vec{x} = u\vec{h}_1 + (1\text{-}u)\,\vec{h}_2 + \vec{\varepsilon}$

$Classe2$: $\vec{x} = u\vec{h}_1 + (1\text{-}u)\,\vec{h}_3 + \vec{\varepsilon}$

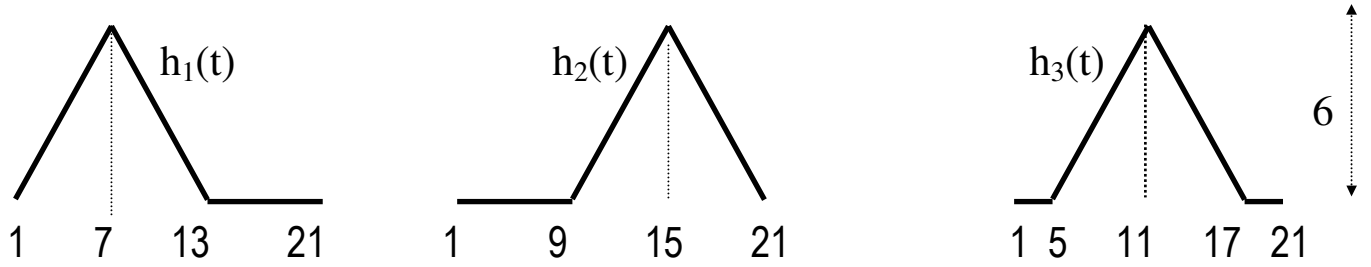
$Classe3$: $\vec{x} = u\vec{h}_2 + (1\text{-}u)\,\vec{h}_3 + \vec{\varepsilon}$

$0 \le u \le 1$: variable aléatoire de distribution uniforme.

$\vec{\varepsilon}$: bruit gaussien de distribution **N**(0,1)

14% : Borne inférieure à l'erreur de généralisation (Bayes).

Ondes de base $h_1(t)$, $h_2(t)$ et $h_3(t)$ t=1,...,21

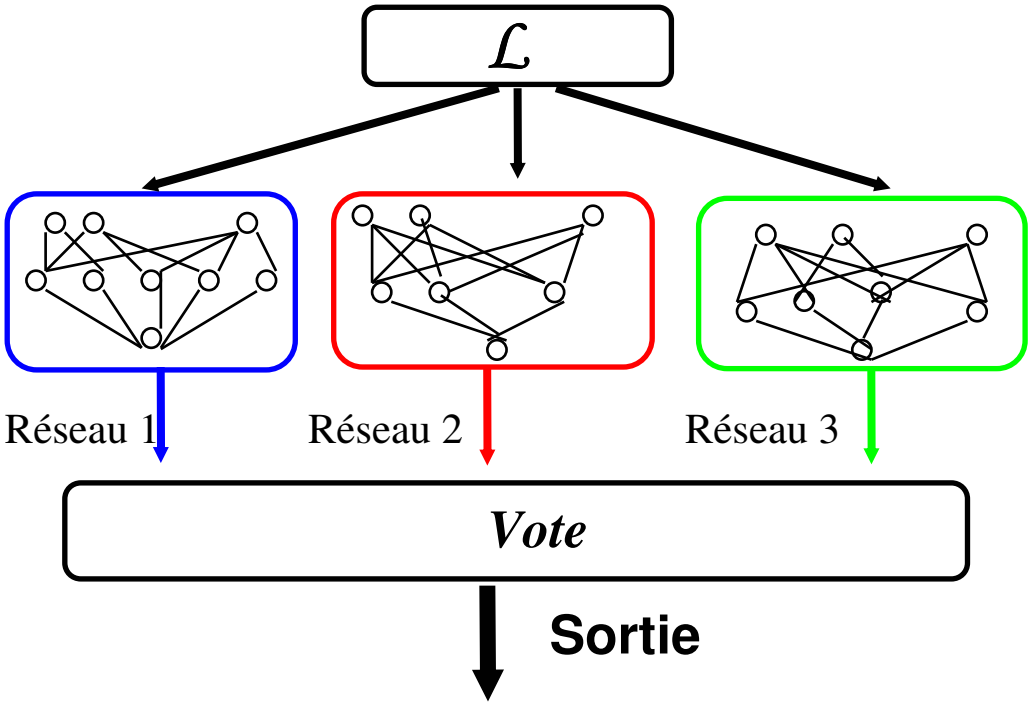


Apprentissage

N	21 valeurs $\in \Re$
P	300
Classes	{1,2,3}
Distribution	$\approx$ 33% par classe
Ensembles	10 bases

Généralisation

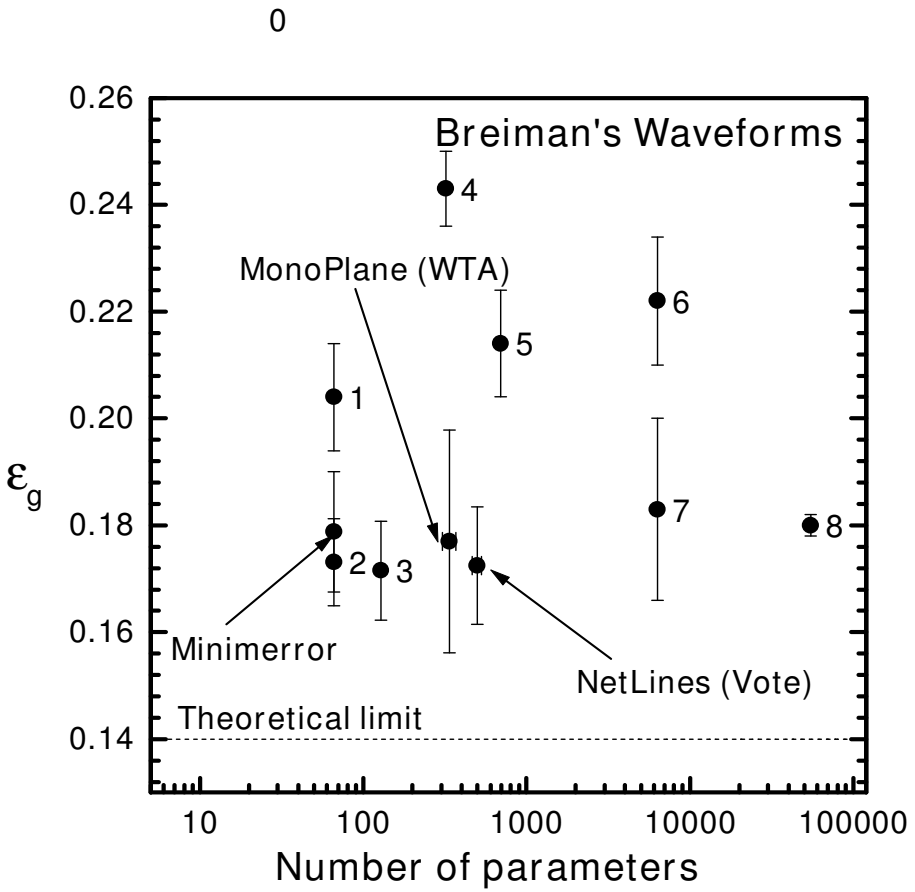
Taille de la base	$G = 5000$
Type de test	*Holdout*



Solution

- 3 réseaux, dédiés à la séparation d'une classe par rapport aux deux autres
- Classe attribuée à chaque exemple par *Vote*

Formes d'ondes de Breiman♣



Méthode*

1 Disc. linéaire
2 Standard du Perceptron
3 Retropropagation
4 Algorithme génétique
5 Disc. quadratique
6 Fenêtres de Parzen
7 K plus proches voisins
8 Constraint

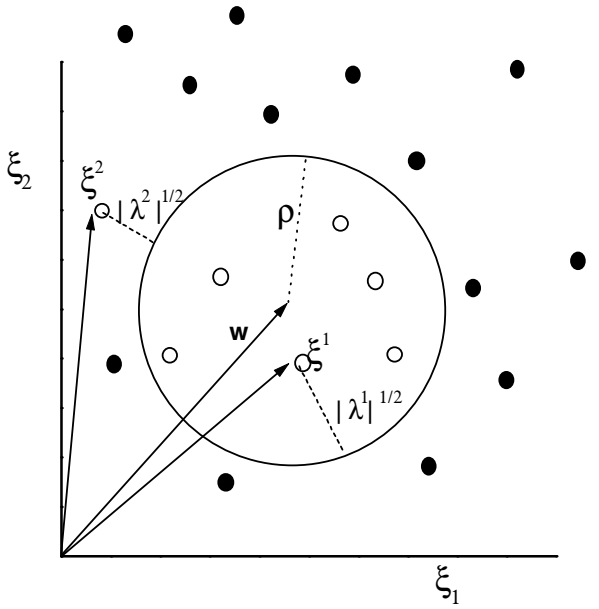
♣Neural Computation (1997) à paraître.

- **11 ensembles:** SYMENU (article collectif, O. Gascuel coordonateur des travaux) 5èmes Journées Nationales du PRC-IA (Nancy), Teknea, 29-76, 1995.
- **Données en ftp://blanche.polytechnique.fr/pub/Symenu/Bases/**

Perspectives...

Minimerror-S

- Même fonction de coût ***E***
- Même type de recuit déterministe.
- Stabilité sphérique λ :



$$\lambda^{\mu} = \left((\vec{\xi} - \vec{\mathbf{w}})^2 - \rho^2 \right) \tau^{\mu}$$

$\lambda^1 > 0 \; ; \; \lambda^2 < 0$

Algorithme NetSphères

- Même heuristique que **NETLINES**
- Entraînement par **Minimerror-S**
- Neurones cachées ***sphériques***
- Neurone de sortie linéaire

CONCLUSION

- Présentation de l'algorithme MINIMERROR :
  Perceptrons binaires,
  Recuit déterministe + descente en gradient
- Deux nouveaux algorithme constructifs :
  MONOPLAN (entrées binaires) et NETLINES (entrées réelles)
- Tests sur nos algorithmes : réalistes et académiques
- Perspectives : réseaux de neurones hybrides
  (hypersphères + hyperplans)

Si l'homme est neuronal, le neurone lui, est inhumain