EMD IA Master II SID

Exo 1 :

1) Donner la règle d'apprentissage de rétro-propagation pour un neurone dans une couche cachée pour une fonction d'activation logsigmoidal. (1pt)
2) Citez les différences entre la règle de la rétropropagation du Gradient et la règle d'apprentissage Delta. Dans quelle cas les deux règles sont similaires ? (3pt)
3) Soit la fraction d'un RNA donné par la Figure 1. Calculer la mise à jour des poids du neurone 1 encadré (w11, w21, b1) en utilisant les résultats de la phase de propagation avant donné sur la figure et un µ=1 considérez des fonctions log sigmoïdal pour tous les neurones. (6pt)

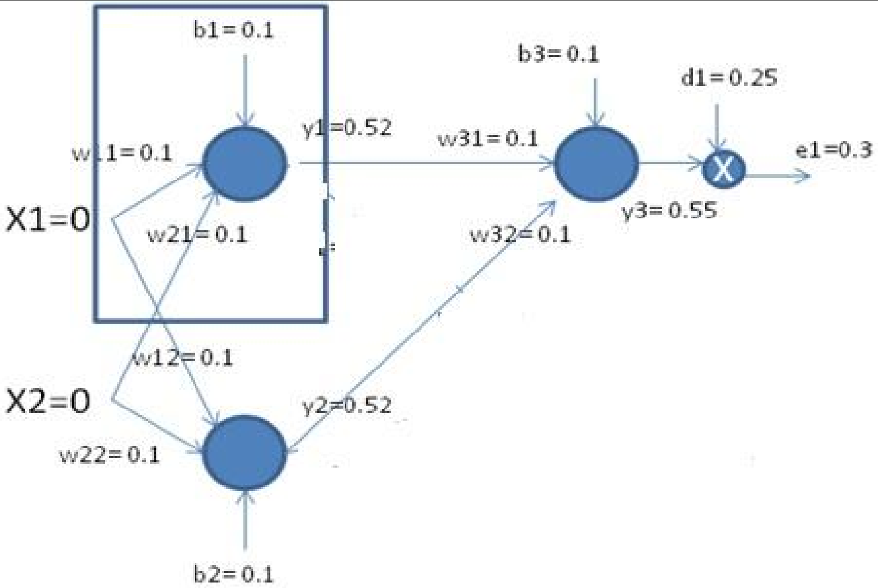


Fig1 : PMC à 2 couches et 2 sorties

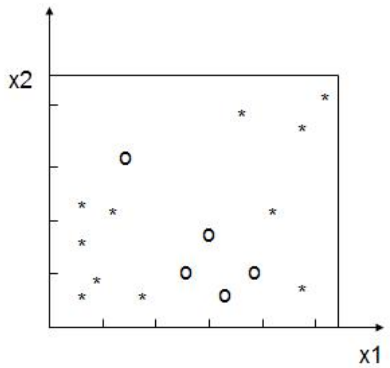


Fig 2.

Exo 2 :

1) Quelle est la nature de la séparation pour un problème de classification à 2 dimensions données pas un RNA Fig 2.
2) Est il possible de résoudre le problème de classification Fig2 en utilisant un neurone de type Perceptron? Pourquoi? Sinon combien de neurones faut il?
3) Donner la forme réseau qui résout le problème Fig 2. Calculer les poids et les biais d'un seul neurone par une méthode graphique

ELEMENTS DE REPONSES

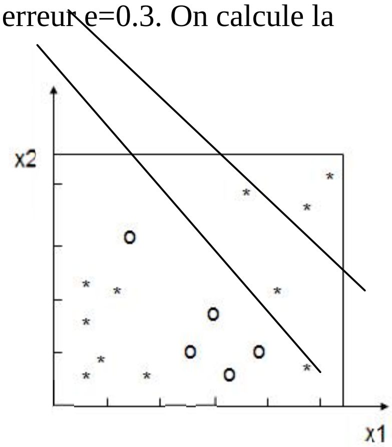
Exo1 :

1. Equation (3.31) pages 35.
2. - La règle de la retropropagation du Gradient est applicable au Réseaux de neurones alors que la règle Delta est applicable aux neurones simples.

- La rétropropagation est applicables a plusieurs sorties alors que la règle Délta juste à une sortie.
- Les deux règles sont similaires dans le cas d'un neurone de sortie, la généralisation de la règle Délta donne la rétropropagation du Gradient.

3. Sur la base du calcul de Y_3 qui prend comme entrée Y_1 et Y_2 et de l'erreur e=0.3. On calcule la mise à jour des poids W3, comme suit :

$$\begin{bmatrix} w_{31}^1 \\ w_{32}^1 \\ b_3^1 \end{bmatrix} = \begin{bmatrix} w_{31}^0 \\ w_{32}^0 \\ b_3^0 \end{bmatrix} + \mu\, e\, Y_3(1 - Y_3) \begin{bmatrix} y_1 \\ y_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix} + 0.074 \begin{bmatrix} 0.52 \\ 0.52 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.13 \\ 0.13 \\ 1.1 \end{bmatrix}$$



Une fois la mie à jour des poids du neurone 3 calculés, o peut etre à jour les poids du neurone 1 comme suit :

$$\begin{bmatrix} w_{11}^1 \\ w_{21}^1 \\ b_1^1 \end{bmatrix} = \begin{bmatrix} w_{11}^0 \\ w_{21}^0 \\ b_1^0 \end{bmatrix} + \mu\; Y_1(1 - Y_1) e\, w_{31}^1 Y_3(1 - Y_3) \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix} + 0.024 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1024 \end{bmatrix}$$

Exo2 :

1. Le problème présenté Fig 1. Est non linéairement séparable.
2. Il faut 3 Neurones pour résoudre le problème. 2 neurones dans la couche caché et un neurone de sortie. Voir Fig 3.5 pages 35 du support de cours.
3. La figure montre l'emplacement des deux droites séparatrices données par 2 neurones de la couche cachée, l'équation donnée par le neurone 1 par exemple est :

$$x_2 = -\frac{w_1}{w_2}\, x_1 - \frac{b_1}{w_2}$$

Il faut deux points pour trouver une solution :

$x_1 = 0,\ x_2 = 3.5$; alors si $w_2 = 1$ *alors* $b_1 = -3.5$.

$$x_2 = 0, \qquad x_1 = 3.0;\ alors\ w_1 = -\frac{3.1}{3.5} = 0.9.$$