# Rapport Projet d’Informatique : Perceptron Multi-Couches

**Valentin Ponzi et Marco Bousseau**

## Sujet :

L’objectif du projet est de créer un Perceptron Multi-Couches. Nous avons choisit un Perceptron qui reconnait un “motif” dessiné sur une plaque de 4 cases formant un carré. Le réseau neuronal renvoie donc un chiffre correspondant a un motif. Il est alors nécessaire de définir des motifs :

- la plaque “*vide*” : aucune case de la plaque n’est pleine;
- le “*point*” : une case de la plaque est pleine;
- la “*ligne*” : deux cases adjacente de la plaque sont pleines;
- la “*diagonale*” : deux cases non-adjacentes de la plaque sont pleines;
- le “*elle*” : trois cases de la plaque seulement sont pleines (forme de L);
- le “*plein*” : toutes les cases de la plaque sont pleines.

Remarque notation : Pour la suite, le neurone m (croissant de haut en bas en partant de 1 pour le premier neurone), de la couche n (croissant de gauche a droite en partant de 1 pour la couche d’entrée) sera noté : $n_{n,m}$

## Couche d’entrée :

La couche d’entrée est composée de 4 neurones qui correspondent chacun à une case du tableau. L’ordre d’entrée est défini comme tel : le premier neurone correspond à la case en haut à gauche de la plaque, le deuxième neurone correspond à la case en haut à droite, le troisième neurone correspond à la case en bas à gauche et le quatrième neurone à la case en bas à droite. Une case vide a une valeur d’activation de 0 et une case pleine a une valeur d’activation de 1.

## Couche cachée 1 et coefficients synaptiques associés :

### Fonctions d’activation et neurones

L’objectif de la couche cachée 1 est d’attribuer à chaque possibilité d’activation de la plaque un neurone qui sera alors activé. On a donc le premier neurone qui correspond à l’activation du “*plein*” : si toutes les cases de la plaque sont pleines alors ce neurone a pour activation 1, sinon 0. On utilisera alors la fonction d’activation suivante :

$$f_{F4}(x) = floor(\frac{x}{4})$$

où x est la somme pondérée des valeurs d’activation des neurones de la couche d’entrée.

De la même façon, les deuxièmes et troisièmes neurones de la couche cachée numéro 1 correspondent a l’activation des deux “*diagonales*”. Le deuxième neurone représente la diagonale haut-gauche/bas-droite et le troisième avec la diagonale haut-droite/bas-gauche. La fonction d’activation de ces deux neurones est la suivante :

$$f_{F2}(x) = floor(\frac{x}{2})$$

où x est la somme pondérée
des valeurs d'activation
des neurones de
la couche d'entrée.

Ensuite, les neurones 4, 5, 6, et 7 de la première couche cachée correspondent à l'activation respective des cases haut-gauche, haut-droite, bas-gauche, et bas-droite. On y représente donc le motif "*point*". La fonction d'activation est notée :

$$f_{=}(x) = x$$

où x est la somme pondérée
des valeurs d'activation
des neurones de
la couche d'entrée.

Ensuite, les neurones 8, 9, 10, et 11 représentent l'activation des "*lignes*" respectives du haut, de la droite, du bas et de la gauche. Leur fonction d'activation est la suivante :

$$f_{F2}(x) = floor(\frac{x}{2})$$

où x est la somme pondérée
des valeurs d'activation
des neurones de
la couche d'entrée.

Enfin, les neurones 12,13,14, et 15 représentent l'activation commune des "*elle*" respectifs des coins haut-droite, bas-droite, bas-gauche, et haut-gauche. La fonction d'activation est alors :

$$f_{F3}(x) = floor(\frac{x}{3})$$

où x est la somme pondérée
des valeurs d'activation
des neurones de
la couche d'entrée.

## Coefficients synaptiques

Les coefficients synaptiques ne sont en théorie pas a préciser dans la rédaction initiale du Perceptron. Cependant, et étant donne qu'il n'est pas du rôle de l'examinateur d'entrainer le Perceptron, nous avons décidé d'attribuer les bons coefficients synaptiques dès la création du réseau. Nous allons ici developper la répartition des coefficients synaptiques qui lient la couche d'entrée a la couche cachée 1. Dans tous les cas prochains, le système est le même : on attribue un coefficient de 1 au neurones de la couche d'entrée dont on veut vérifier l'activation dans le neurone d'"arrivée".

La pondération de chacun des neurones $n_{1,i}$ $(1 \leq i \leq 4)$ , dans le calcul de la valeur d'activation du neurone $n_{2,1}$ vaut 1. En effet, on veut vérifier que tous les neurones $n_{1,i}$ $(1 \leq i \leq 4)$ sont activés. S'ils le sont tous, alors $f_4(x) = 1$ et la valeur d'activation de $n_{2,1}$ est 1; sinon elle est de 0. La matrice de ces coefficients est : $\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$.

Selon le même système, les coefficients synaptiques du neurone $n_{2,2}$ sont les suivants : $\begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$.

On extrapole aux autres neurones $n_{2,i}(1 \leq i \leq 11)$ :

$$n_{2,3} \to \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \quad n_{2,4} \to \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad n_{2,5} \to \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad n_{2,6} \to \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad n_{2,7} \to \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad n_{2,8} \to \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$n_{2,9} \to \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \; n_{2,10} \to \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \; n_{2,11} \to \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} \; n_{2,12} \to \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} \; n_{2,13} \to \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} \; n_{2,14} \to \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} \; n_{2,15} \to \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

On obtient alors la matrice de coefficients synaptiques suivante :

$$Coeff1 = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

# Couche cachée 2 et coefficients synaptiques associés :

## Fonction d’activation et neurones

La deuxième couche cachée sert à compter le nombre d’occurrence de chaque type. Le premier neurone est donc le même que celui de la deuxième couche cachée. On utilisera donc la fonction d’activation définie plus haut : $f_{=}(x)$. Le deuxième neurone compte le nombre de diagonales de la plaque. Sa fonction d’activation est donc : $f_{=}(x)$. Le troisième neurone compte le nombre de points de la plaque. Sa fonction d’activation est aussi : $f_{=}(x)$. Le quatrième neurone compte le nombre de lignes de la plaque. Sa fonction d’activation est donc aussi : $f_{=}(x)$. Le quatrième neurone compte le nombre de “*elle*” de la plaque. Sa fonction d’activation est la même que celle des autres neurones : $f_{=}(x)$.

### Coefficients synaptiques

Chaque neurone de la couche cachée 2 est la somme des activations des neurones qui vérifient les activations de ces types, sur la couche cachée 1. On donne donc la matrice de coefficients synaptiques suivante : $Coeff2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$.

## Couche de sortie et coefficients synaptiques associés :

### Fonction d'activation et neurone

La couche de sortie ne comporte qu'un seul neurone qui aura pour valeur d'activation le "résultat" du Perceptron. Ce neurone contient la somme des valeurs d'activation des neurones de la couche cachée 2. Sa fonction d'activation est donc : $f_{=}(x)$.

### Coefficients synaptiques

Les coefficients synaptique de passage de la couche cachée 2 a la couche de sortie on fait l'objet d'une raisonnement arbitraire : il nous fallait en effet distinguer le cas d'une "*diagonale*" de celui d'une "*ligne*".En effet, une ligne est composée de 2 "*point*" et de 1 "*ligne*" donc le résultat renvoyé par le Perceptron (dans le cas de coefficients synaptiques de 1) est : 2*1 + 1*1 = 3. Dans le cas d'une diagonale, on a aussi 2 "*point*" et 1 "*diagonale*", donc avec le même système de coefficients synaptiques, le résultat renvoyé par le Perceptron est : 2*1 + 1*1 = 3. On avait alors deux résultats égaux pour deux motifs différents. Nous avons donc décidé de pondérer la synapse qui lie le type "*diagonale*" avec le neurone de sortie avec un coefficient de 2 plutôt que 1. Ainsi, une "*ligne*" donne toujours 3 et une diagonale donne : 2*1 + 1*2 = 4. La matrice de coefficients synaptiques est alors :

$$Coeff3 = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

# Resultat du perceptron

Il y a 6 types différents et 6 issues possible au Perceptron. On associe donc à chaque motif un chiffre qui doit être renvoyé par le Perceptron.

Vide : si la plaque est vide, tous les neurones de la couche cachée 1 et de la couche cachée 2 ont pour valeur d'activation 0. Le chiffre renvoyé est donc 0.

Point : si un seul point de la plaque est plein, alors seul le neurone $n_{2,i}(4 \leq i \leq 7)$ de la couche cachée 1 a une valeur d'activation de 1. Ainsi, seul le neurone $n_{3,3}$ de la couche cachée 3 a une valeur d'activation de 1. Donc la valeur renvoyé est : 1*1 = 1.

Ligne : si une ligne est activée, alors il y a 2 points et 1 ligne d'actives et la valeur renvoyée est : 2*1 + 1*1 = 3.

Diagonale : si une diagonale est activée, alors il y a 2 "*point*" d'activés ,1"*diagonale*". La valeur renvoyée est donc : 2*1 + 1*2 = 4.

Elle : si un "elle" est activé, alors il y a 3 "*point*" d'activés , 2 "*ligne*", 1 "*diagonale*", et 1 "*elle*". La valeur renvoyée est donc : 3*1 + 2*1 + 1*2 + 1*1 = 8.

Plein : si toute les cases sont activées, alors il y a 4 "*point*" d'activés , 4 "*ligne*", 1 "*plein*", 2 "*diagonale*", et 4 "*elle*". La valeur renvoyée est donc : 4*1 + 4*1 + 2*2 + 4*1 = 16.

# Descente de Gradient

Il n'est pas pertinent ici de faire une descente de gradient étant donné le fait qu'on ait utilise la fonction $floor$ qui n'est pas dérivable. On peut en revanche approximer la fonction $floor$ avec la fonction : $f_{=}(x) = x$. Ainsi, on a une correspondance entre les fonctions effectivement utilisées et des fonctions que l'ont utilise pour la descente de gradient.

$$f_{F4}(x) = floor(\frac{x}{4}) \ \rightarrow \ f_4(x) = \frac{x}{4} \ \Rightarrow \ f_4'(x) = \frac{1}{4}$$

$$f_{F2}(x) = floor(\frac{x}{2}) \ \rightarrow \ f_2(x) = \frac{x}{2} \ \Rightarrow \ f_2'(x) = \frac{1}{2}$$

$$f_{F3}(x) = floor(\frac{x}{3}) \ \rightarrow \ f_3(x) = \frac{x}{3} \ \Rightarrow \ f_3'(x) = \frac{1}{3}$$

Ainsi, la dérivée de l'erreur par rapport au coefficient synaptique est :

$$\frac{\partial E}{\partial \alpha_j} = -(r_{att} - r) \cdot f_k'(\sum_{i=1}^{n} \alpha_i \cdot x_i) \cdot xj = \frac{-(r_{att} - r) \cdot x_j}{k} = \frac{e_j^l}{k}$$

La descente de gradient est alors possible.