Les Réseaux de Neurones

Philibert PAPPENS, Tristan POIDATZ, Mathurin PETIT

2021-2022

Contents

1 Définitions

1.1 Neurone

Un Neurone est une représentation mathématique et informatique d'un neurone biologique. Il contient en général plusieurs entrées et une seule sortie. Mathématiquement, un neurone est une fonction à plusieurs variables et à valeurs réelles.

1.2 Réseau de Neurones

Un réseau de Neurones est une association de neurones pour accomplir des tâches arbitrairement complexes.

1.3 Fonction d'activation

Une fonction d'activation (souvent notée σ) est une fonction de $\mathbb{R}^{\mathbb{R}}$ dont son calcul et celui de sa dérivée est peu couteux en temps. Choisir des fonctions d'activations non-linéaires permet au réseau de créer des comportements plus complexes.

2 Notations

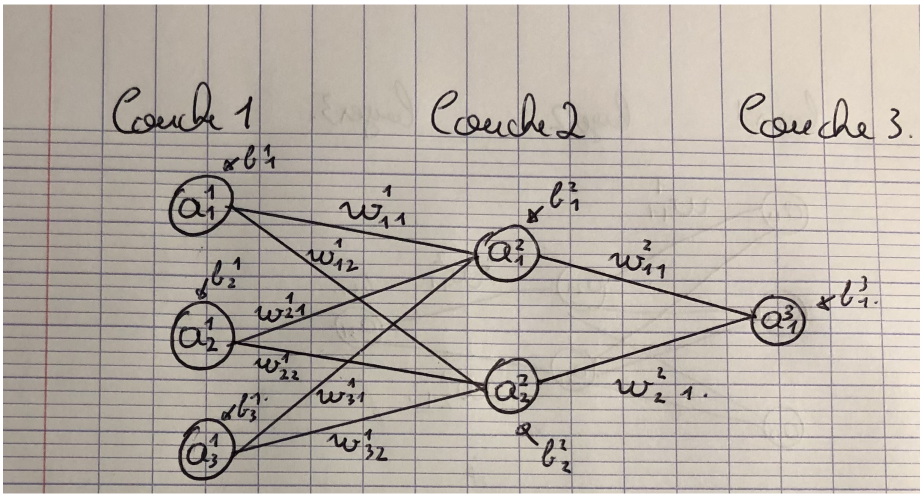

Figure 1: Réseau simple à trois couches

2.1 Poids

$$w_{i,j}^{\ell}$$

est le poids reliant le neurone i de la couche ℓ au neurone j de la couche $\ell + 1$

2.2 Activation intermédiaire

Considérons une couche $\ell \in \mathbb{N}$ d'un réseau. Notons n le nombre de neurones dans la couche ℓ. Pour le neurone i de la couche $\ell + 1$, on définit sa valeur d'activation intermédiaire:

$$z_i^{\ell+1} = \sum_{k=1}^{n} w_{k,i}^{\ell} a_k^{\ell} \tag{1}$$

2.3 Biais

$$b_i^{\ell}$$

est la valeur du biais i de la couche l.

2.4 Fonction d'activation

$$\sigma^{\ell} :$$

est la fonction d'activation permettant de passer de la couche $\ell - 1$ à la couche ℓ.

2.5 Activation

$$a_i^{\ell}$$

est la valeur du neurone i de la couche ℓ. Elle vaut $\sigma(z_i^{\ell})$

2.6 Matrice d'activation

$$A_\ell = \begin{pmatrix} a_1^\ell & a_2^\ell & ... & a_n^\ell \end{pmatrix}$$

2.7 Matrice d'activations intermédiaires

$$Z_\ell = \begin{pmatrix} z_1^\ell & z_2^\ell & ... & z_n^\ell \end{pmatrix}$$

2.8 Matrice des poids

$$W_\ell = \begin{pmatrix} w_{1,1}^\ell & w_{1,2}^\ell & ... & w_{1,k}^\ell \\ w_{2,1}^\ell & w_{2,2}^\ell & ... & w_{2,k}^\ell \\ ... & ... & ... & ... \\ w_{n,1}^\ell & w_{n,2}^\ell & ... & w_{n,k}^\ell \end{pmatrix}$$

Où n est le nombre de neurones dans la couche ℓ et k dans la couche $\ell + 1$

2.9 Matrice gradient

Pour une matrice M_ℓ, on note ∇_M^ℓ la matrice des dérivées partielles par rapport a chacun des coefficients. Par exemple:

$$\nabla_A^\ell = \begin{pmatrix} \frac{\partial}{\partial a_1^\ell} & \frac{\partial}{\partial a_2^\ell} & ... & \frac{\partial}{\partial a_n^\ell} \end{pmatrix}$$

2.10 Matrice δ

On pose $\delta^\ell = \nabla_Z^\ell C$

2.11 Produit de Hadamard

L'operateur $\odot$ permet de definir le produit de Hadamard: le produit matriciel terme a terme:

$$\begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix} \odot \begin{pmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{pmatrix} = \begin{pmatrix} a_{1,1} \cdot b_{1,1} & a_{1,2} \cdot b_{1,2} \\ a_{2,1} \cdot b_{2,1} & a_{2,2} \cdot b_{2,2} \end{pmatrix}$$

3 Formules de propagation

3.1 Propagation avant

L'équation (1) et la définition de Z_ℓ peuvent être mises sous forme matricielle pour donner:

$$Z_{\ell+1} = A_\ell \cdot W_\ell + B_{\ell+1} \tag{2}$$

$$A_{\ell+1} = \sigma(Z_{\ell+1}) \tag{3}$$

3.2 Propagation arrière

Par le théorème de dérivation des fonctions composées, on a:

$$\frac{\partial C}{\partial z_j^\ell} = \frac{\partial C}{\partial a_j^\ell} \cdot \frac{\partial a_j^\ell}{\partial z_j^\ell} = \frac{\partial C}{\partial a_j^\ell} \sigma'(z_j^\ell)$$

Ce qui permet d'obtenir une première equation de rétro-propagation:

$$\delta^\ell = \nabla_a^\ell C \odot \sigma'(Z_\ell) \tag{4}$$

Puis,

$$\nabla_a^\ell C = \delta^{\ell+1} \cdot (W_\ell)^T \tag{5}$$

Les equations (4) et (5) permettent d'établir une relation de récurrence descendante sur la suite $(\delta^\ell)_{\ell \in \mathbb{N}}$:

$$\delta^\ell = (\delta^{\ell+1} \cdot (W_\ell)^T) \odot \sigma'(Z_\ell) \tag{6}$$

En appliquant une seconde fois ce théorème, on effectue le calcul matriciel suivant:

$$\begin{aligned}
\nabla_w^\ell C &= \begin{pmatrix} \frac{\partial C}{\partial w_{1,1}^\ell} & \frac{\partial C}{\partial w_{1,2}^\ell} & \dots & \frac{\partial C}{\partial w_{1,k}^\ell} \\ \frac{\nabla C}{\partial w_{2,1}^\ell} & \frac{\partial C}{\partial w_{2,2}^\ell} & \dots & \frac{\partial C}{\partial w_{2,k}^\ell} \\ \dots & \dots & \dots & \dots \\ \frac{\partial C}{\partial w_{n,1}^\ell} & \frac{\partial C}{\partial w_{n,2}\ell} & \dots & \frac{\partial C}{\partial w_{n,k}\ell} \end{pmatrix} \\
&= \begin{pmatrix} \frac{\partial C}{\partial z_1^{\ell+1}} \cdot \frac{\partial z_1^{\ell+1}}{\partial w_{1,1}^\ell} & \frac{\partial C}{\partial z_2^{\ell+1}} \cdot \frac{\partial z_2^{\ell+1}}{\partial w_{1,2}^\ell} & \dots & \frac{\partial C}{\partial z_k^{\ell+1}} \cdot \frac{\partial z_k^{\ell+1}}{\partial w_{1,k}^\ell} \\ \frac{\partial C}{\partial z_1^{\ell+1}} \cdot \frac{\partial z_1^{\ell+1}}{\partial w_{2,1}^\ell} & \frac{\partial C}{\partial z_2^{\ell+1}} \cdot \frac{\partial z_2^{\ell+1}}{\partial w_{2,2}^\ell} & \dots & \frac{\partial C}{\partial a_k^{\ell+1}} \cdot \frac{\partial z_k^{\ell+1}}{\partial w_{2,k}^\ell} \\ \dots & \dots & \dots & \dots \\ \frac{\partial C}{\partial z_1^{\ell+1}} \cdot \frac{\partial z_1^{\ell+1}}{\partial w_{n,1}^\ell} & \frac{\partial C}{\partial z_2^{\ell+1}} \cdot \frac{\partial z_2^{\ell+1}}{\partial w_{n,2}^\ell} & \dots & \frac{\partial C}{\partial z_k^{\ell+1}} \cdot \frac{\partial z_k^{\ell+1}}{\partial w_{n,k}^\ell} \end{pmatrix} \\
&= \begin{pmatrix} \nabla_z^{\ell+1} C \cdot a_1^\ell \\ \nabla_z^{\ell+1} C \cdot a_2^\ell \\ \dots \\ \nabla_z^{\ell+1} C \cdot a_n^\ell \end{pmatrix} \\
&= \begin{pmatrix} \delta^{\ell+1} \cdot a_1^\ell \\ \delta^{\ell+1} \cdot a_2^\ell \\ \dots \\ \delta^{\ell+1} \cdot a_n^\ell \end{pmatrix} \\
&= \begin{pmatrix} \delta^{\ell+1} \\ \delta^{\ell+1} \\ \dots \\ \delta^{\ell+1} \end{pmatrix} \odot (A_\ell)^T
\end{aligned}$$

On obtient ainsi:

$$\nabla_w^\ell C = \begin{pmatrix} \delta^{\ell+1} \\ \delta^{\ell+1} \\ \dots \\ \delta^{\ell+1} \end{pmatrix} \odot (A_\ell)^T \tag{7}$$

Par un calcul analogue, on obtient:

$$\nabla_b^\ell C = \delta^\ell \tag{8}$$

4 Descente de gradient

4.1 Principe