



## TD 6 – Optimisation Réseaux de neurones

▷ **Exercice 1.** On s'intéresse ici à la modélisation via un neurone formel.

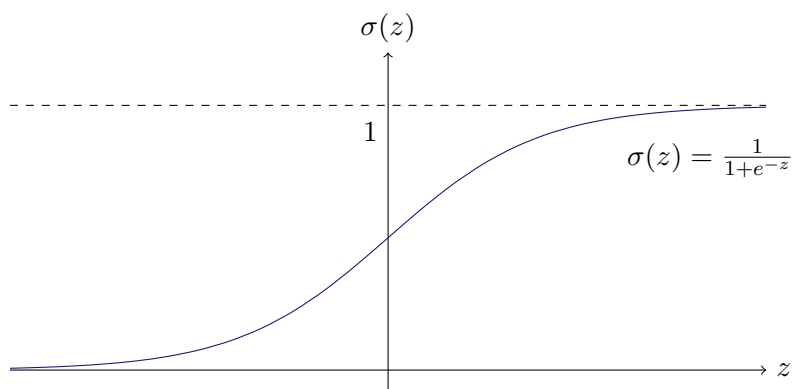
**Définition 1.** Un neurone formel est une fonction paramétrée par  $n+1$  paramètres  $w_1, \dots, w_n, b$  :

$$\begin{aligned} g: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} &\longrightarrow \mathbb{R} \\ (x, w, b) &\longmapsto g(x, w, b) := \sigma(\sum_{i=1}^n w_i x_i + b) \end{aligned}$$

où  $\sigma$  est une fonction donnée qui s'appelle une fonction d'activation. Chaque paramètre  $w_i$  s'appelle le poids synaptique associé au signal d'entrée  $x_i$ .

On prendra dans la suite, sauf mention contraire, comme fonction  $\sigma$  la fonction sigmoïde :

$$\begin{aligned} \sigma: \mathbb{R} &\longrightarrow \mathbb{R} \\ z &\longmapsto \sigma(z) := \frac{1}{1 + e^{-z}}. \end{aligned}$$



La figure 1 schématise un neurone formel.

1. produit scalaire entre les entrées  $x$  et les poids synaptiques  $w$  :  $w^T x$  ;
2. ajout d'une valeur de référence (biais  $b$ ) :  $z = w^T x + b$
3. application de la fonction d'activation à la valeur obtenue  $z$  :  $a = \sigma(z)$

**Définition 2.** On a à notre disposition  $K$  points  $x^k \in \mathbb{R}^n$  et  $y^k \in \mathbb{R}$ , on appelle apprentissage du neurone l'estimation par les moindres carrés des paramètres du neurone.

**1.1.** Écrire le problème aux moindres carrés qui définit l'apprentissage. On donnera en particulier la fonction résidu  $r$  en précisant clairement l'espace de départ et l'espace d'arrivée.

**1.2.** Calculer la dérivée de la fonction à minimiser  $f(\beta)$ .

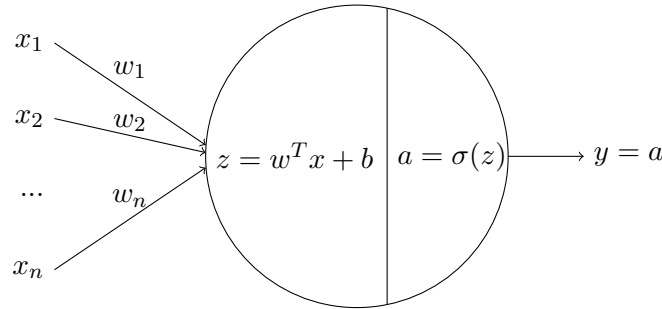
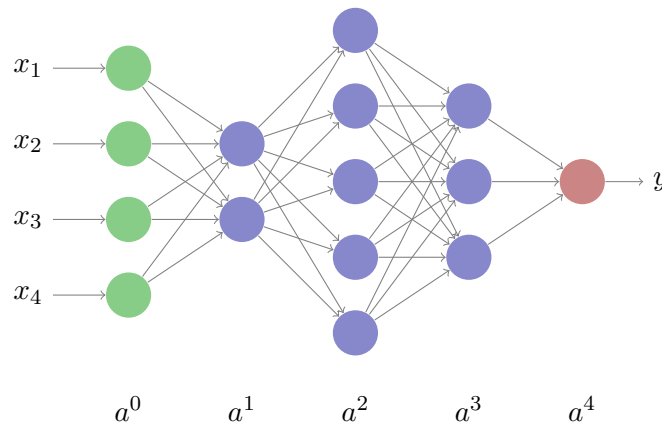


FIGURE 1 – Représentation schématique d'un neurone.

FIGURE 2 – Schéma d'un réseau avec une couche en entrée ( $a^0 = x$ ), 3 couches cachées et une couche en sortie.

▷ **Exercice 2.** On considère maintenant le cas d'un réseau à plusieurs couches données par le schéma de la figure 2.

**2.1.** On note  $L$  le nombre de couches (sans compter la couche d'entrée), donnez :

- les nombre de neurones  $n_l$  pour  $l = 1, \dots, L$  dans chaque couche et la dimension  $n_0 = n$  des données en entrée ;
- Les dimensions des paramètres  $W^l$  et  $b^l$  intervenant dans la couche  $l$ .

**2.2.** On note  $a^l$  la sortie de la couche  $l$  ( $a^l$  s'appelle dans la terminologie des réseaux de neurones l'activation de la couche  $l$ ). Écrire  $a^{l+1}$  en fonction de  $a^l$ ,  $W^{l+1}$ ,  $b^{l+1}$  et de  $\sigma$ . En déduire la fonction exprimant  $y$  en fonction de  $x$  et des paramètres du modèle :  $y(x, \beta)$ .

**2.3.** On suppose que l'on a comme données  $K$  couples  $(x^k, y^k)_{k=1, \dots, K}$ , où  $x^k \in \mathbb{R}^n$  et  $y^k \in \mathbb{R}$ . Écrire dans ce cas le problème aux moindres carrés qui définit l'apprentissage.

**2.4.** Calculer les dérivées partielles de  $y(x, \beta)$  par rapport à  $\beta^4 = (W^4, b^4)$  et par rapport à  $\beta^3$ . En déduire les dérivées partielles de la fonction à minimiser par rapport à ces paramètres  $\beta^4$  et  $\beta^3$ .

**Remarque 1.** En pratique :

- la sortie  $y$  est dans  $\mathbb{R}^{n_L}$  ;
- On n'utilise pas les moindres carrés, mais une autre fonction de coût ;