Feuille 4 : réseaux de neurones

Rappel sur les réseaux de neurones. On considère un réseau de neurones avec plusieurs couches cachées, de données d'entrée $x = (x_1, \dots, x_n)$ et de donnée de sortie z (une seule sortie, pour simplifier).

Le réseau est défini couche par couche avec M couches, $k = M$ étant la couche de sortie. La couche d'entrée est représentée par $k = 0$. Soit la couche $k \in [\![1, M]\!]$ du réseau comportant p neurones et la couche $(k-1)$ en comportant q. Sur la couche k, on note :

- W^k la matrice des poids de la couche $(k-1)$ vers la couche k : w^k_{ij} est le poids du neurone j de la couche $(k-1)$ vers le neurone i de la couche k. La matrice W^k est de taille (p, q).
- b^k le vecteur des biais des neurones de la couche k (vecteur de taille p).
- la même fonction d'activation ϕ^k pour tous les neurones de la couche k (sauf ceux de la couche d'entrée qui n'en ont pas).

Pour un neurone $i \in [\![1, p]\!]$ de la couche $k \in [\![1, M]\!]$, on définit la fonction d'évaluation du neurone :

$$f^k_i(y) = \phi^k\left(b^k_i + w^k_{i1} y_1 + \cdots + w^k_{iq} y_q\right) \tag{1}$$

pour $y = (y_1, \dots, y_q)^\top \in \mathbb{R}^q$.

Evaluations sur le réseau. On définit les valeurs $F^k_i(x)$ du réseau associées au neurone i de la couche k par récurrence à partir des valeurs de la couche précédente $(k-1)$:

- Pour les données d'entrée $x \in \mathbb{R}^n$ (avec $k = 0$) :

$$F^0(x) = \left(F^0_1(x), \dots, F^0_n(x)\right)^\top = x = (x_1, \dots, x_n)^\top \tag{2}$$

- Pour $k \in [\![1, M]\!]$ et $i \in [\![1, p]\!]$,

$$F^k_i(x) = f^k_i\left(F^{k-1}_1(x), \dots, F^{k-1}_q(x)\right) \tag{3}$$

Vectoriellement, on note

$$a^k := F^k(x) = \left(F^k_1(x), \dots, F^k_p(x)\right)^\top \in \mathbb{R}^p \tag{4}$$

et la relation de récurrence (3) s'écrit (pour $k \in [\![1, M]\!]$) :

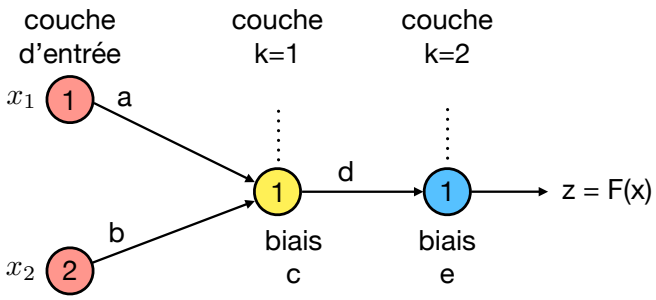
$$\begin{array}{rcl} \alpha^k & = & b^k + W^k a^{k-1} \\ a^k & = & \varphi^k(\alpha^k) \end{array} \tag{5}$$

avec $\alpha^k = \left(\alpha^k_1, \dots, \alpha^k_p\right)^\top \in \mathbb{R}^p$, $\varphi^k(\alpha^k) = \left(\phi^k(\alpha^k_1), \dots, \phi^k(\alpha^k_p)\right)^\top \in \mathbb{R}^p$ et en partant des données d'entrée $a^0 = x$.

Les formules de récurrence (5) permettent de calculer la valeur $F(x)$ du réseau à la sortie (couche de sortie M) pour la donnée d'entrée x par $F(x) := F^M(x) = a^M$.

Exercice 1. *Fonction d'évaluation d'un réseau*

On considère le réseau de neurones suivant possédant deux entrées, une couche cachée d'un seul neurone et une couche de sortie d'un neurone. On suppose que les neurones ont tous la mêmes fonction d'activation ϕ (sauf les neurones d'entrée qui n'ont pas de fonction d'activation).

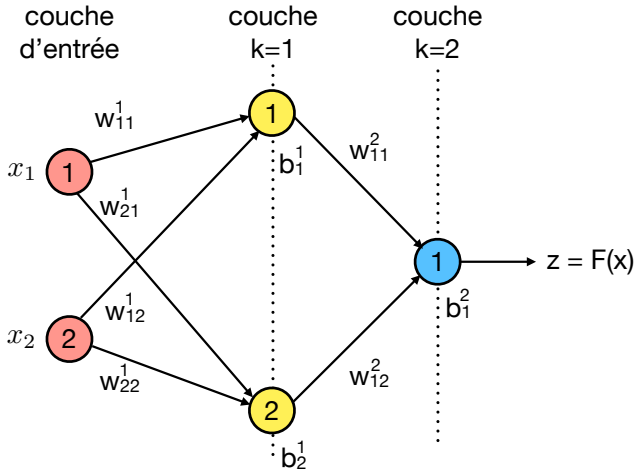


Il y a 5 paramètres à déterminer : les 3 poids (a, b, d) et les 2 biais (c, e).

1. Calculer la fonction d'évaluation F du réseau.
2. Pour les données d'entrée $x = (x_1, x_2)$ et de sortie z, calculer le gradient de l'erreur $E(w) = (F(x) - z)^2$ par rapport aux poids et biais du réseau regroupés dans un vecteur $w = (a, b, c, d, e)^\top$.
3. Ecrire une itération de la méthode de descente de gradient.

Exercice 2. *Rétropropagation du gradient*

On considère le réseau de neurones ci-dessous. Dans cet exercice, on ne va pas calculer le gradient directement à partir de l'expression de F mais par rétropropagation du gradient en partant de la dernière couche vers la première.



Les neurones de la couche cachée ($k = 1$) ont la fonction d'activation sigmoïde σ et la fonction d'activation ϕ pour le neurone de sortie ($k = 2$). Pour une donnée d'entrée $x = (x_1, x_2)$ et de sortie z, on considère l'erreur $E(x) = (F(x) - z)^2$.

1. Ecrire la fonction d'évaluation F du réseau comme une fonction G^2 de α^2, puis comme une fonction G^1 de α^1 :
$$F(x) = G^2(\alpha^2) = G^1(\alpha^1) \tag{1}$$
2. *Calcul du gradient de E par rapport aux poids/biais de la dernière couche de sortie $k = 2$.*
   (a) Calculer la quantité
   $$s^2 = \nabla_{\alpha^2} G^2 = \partial G^2 / \partial \alpha^2_1$$
   (b) Calculer les dérivées $\partial E/\partial w^2_{1j}$ et $\partial E/\partial b^2_1$ en fonction de s^2.
3. *Calcul du gradient de E par rapport aux poids/biais de la couche $k = 1$.*
   (a) A partir de la relation (1), calculer la quantité $s^1 = \nabla_{\alpha^1} G^1$ i.e. les dérivées $s^1_j := \partial G^1/\partial \alpha^1_j$ en fonction de s^2.
   (b) Calculer les dérivées $\partial E/\partial w^1_{ij}$ et $\partial E/\partial b^2_i$ en fonction de s^1.
4. Ecrire l' "algorithme" pour calculer tout le gradient de E.

Exercice 3. *Convergence de l'algorithme du perceptron*

Un perceptron est caractérisé par la fonction $F : \mathbb{R}^n \to \{-1, 1\}$ de la forme $F(x) = \phi(f_w(x))$ avec
$$f_w(x) = (a \,|\, x) + w_0 = \sum_{i=1}^{n} w_i x_i + w_0 \tag{1}$$
et ϕ est une fonction d'activation. On note les poids et biais
$$w = (w_0, a) \in \mathbb{R} \times \mathbb{R}^n.$$
On rappelle (cf. cours) qu'une donnée $(x^{(j)}, z^{(j)}) \in \mathbb{R}^n \times \{-1, 1\}$ est *mal classée* si
$$z^{(j)} f_w(x^{(j)}) < 0 \tag{2}$$
L'algorithme du perceptron (avec gradient stochastique) s'écrit :

— Initialisation à 0 : $k = 0$, $w^{(0)} = (w^{(0)}_0, a^{(0)}) = 0$.
— Tant qu'il existe j tel que $z^{(j)} f_{w^{(k)}}(x^{(j)}) < 0$
- choisir j **aléatoirement** tel que $z^{(j)} f_{w^{(k)}}(x^{(j)}) < 0$.
- $\begin{pmatrix} w_0^{(k+1)} \\ a^{(k+1)} \end{pmatrix} = \begin{pmatrix} w_0^{(k)} \\ a^{(k)} \end{pmatrix} + \eta \begin{pmatrix} z^{(j)} \\ z^{(j)} x^{(j)} \end{pmatrix}$ avec $\eta > 0$
- $k = k + 1$

On va montrer que l'algorithme du perceptron converge (il n'y a plus de donnée mal classée) en un nombre fini d'itérations (Théorème de Novikoff, 1962).

On suppose que :

H1. il existe $\gamma > 0$ et un vecteur $\bar{w} \in \mathbb{R} \times \mathbb{R}^n$ avec $\|\bar{w}\|_2 = 1$ tel que

$$\forall j, \quad z^{(j)} f_{\bar{w}}(x^{(j)}) \geq \gamma \quad \text{(séparabilité linéaire des données)}$$

H2. $\forall j, \quad \|x^{(j)}\|_2 \leq R$ (données bornées)

1. Supposons qu'au bout de k itérations, il existe une donnée $(x^{(j)}, z^{(j)})$ mal classée (et c'est celle qui est choisie par l'algo.). Montrer que $(w^{(k+1)}|\bar{w}) \geq (w^{(k)}|\bar{w}) + \eta\gamma$. En déduire que $(w^{(k+1)}|\bar{w}) \geq k\eta\gamma$ puis à l'aide de Cauchy-Schwarz établir que
$$\|w^{(k+1)}\|_2 \geq k\eta\gamma \tag{3}$$
2. Calculer $\|w^{(k+1)}\|_2^2$ en fonction de $\|w^{(k+1)}\|_2^2$. En utilisant le fait que la donnée $(x^{(j)}, z^{(j)})$ est mal classée, montrer que
$$\|w^{(k+1)}\|_2^2 \leq \|w^{(k+1)}\|_2^2 + \eta^2(1 + R^2)$$
En déduire que
$$\|w^{(k+1)}\|_2^2 \leq k\eta^2(1 + R^2) \tag{4}$$
3. En combinant (3) et (4), montrer que k vérifie nécessairement
$$k \leq (1 + R^2)/\gamma^2 \tag{5}$$
Conclure.