Électif Disciplinaire : Les réseaux de neurones

12 mai 2022

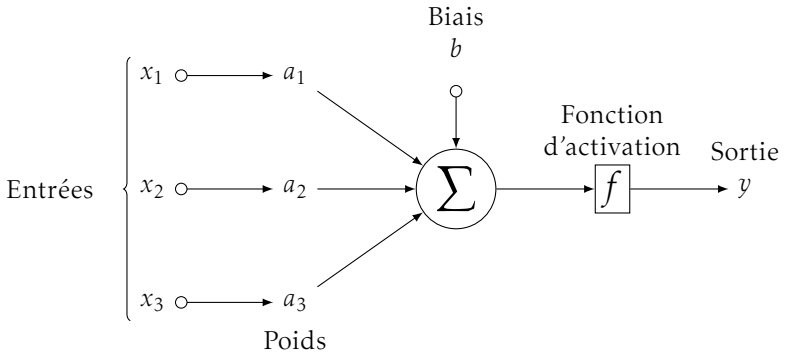
Table des matières

2 Introduction

L'Intelligence Artificielle est une discipline de l'informatique fondamentale qui a pour objectif de simuler le comportement du cerveau humain. Vous avez vu précédemment que les travaux débutent dans les année 1940 avec la notion de neurone formel : c'est la naissance du perceptron. Ensuite, la mise en couche de cet objet permet de simuler des tâche plus complexes comme le fonctionnement de la rétine avec les travaux de Rosenblatt en 1959. Le problème de cette époque est la pluissance de calcul collossale de calcul que cela requiert. Les travaux sont redevenus à la mode avec l'arrivée de supers calculateurs dont la puissance de calcul est colossale à l'heure actuelle.
Le retour en grâce de cette discipline date du début des année 80 avec beaucoup de travaux théorique. Ils retournent au placard au milieu des années 90 toujours faute de puissance de calcul. Enfin, la mise en application dans les années 2000 est permise par l'augmentation des puissance de calcul des ordinateurs. Des applications sont menées tout azimut ; les jeux (go, échecs, starcraft...), le traitement d'image, le traitement du langage...

3 Les réseaux de neurones

On rappelle rapidement la notion de perceptron



Définition 1. *Le perceptron est un modèle abstrait qui est caractérisé par une fonction interne dite d'activation f, il prend en n entrée(s) $x_1,\dots x_n$ et donne une sortie y.*

$$y = f\left(\sum_{k=1}^{n} a_k x_k + b\right)$$

La fonction f permet de transformer le signal d'entrée à l'aide d'une combinaison
La fonction d'activation opère une transformation d'une combinaison affine des signaux d'entrée, a_0, terme constant, étant appelé le biais du neurone.
Cette combinaison affine est déterminée par un vecteur de poids $[a_0,...,a_p]$ associé à chaque neurone et dont les valeurs sont estimées dans la phase d'apprentissage. Ils constituent la mémoire ou connaissance répartie du réseau.
Il existe différentes fonctions d'activation. Les principales sont :

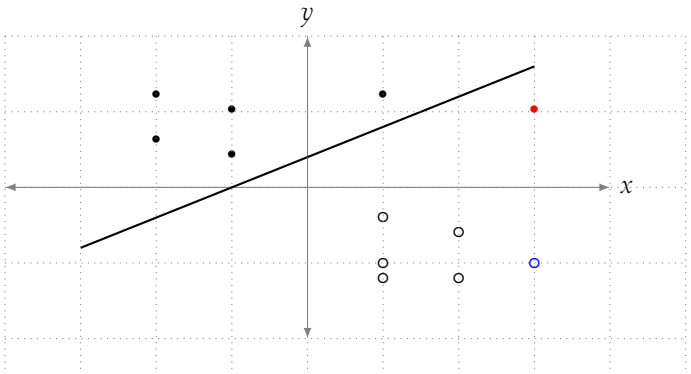
- *La fonction identité*
- *La fonction de Heaviside :* $\mathbb{1}_{[0,+\infty[}$
- *La fonction sigmoïde :* $\dfrac{1}{(1+e^x)}$
- *La fonction ReLU :* $\max(0,x)$
- *La fonction softmax, radiale, stochastique...*

Les fonctions linéaires, sigmoïdes, ReLU, softmax sont bien adaptées aux algorithmes d'apprentissage impliquant une rétro-propagation du gradient car leur fonction d'activation est différentiable ; ce sont les plus utilisées. Le modèle à seuil est sans doute plus conforme à la réalité biologique mais pose des problèmes d'apprentissage (on le verra en TP). Enfin le modèle stochastique est utilisé pour des problèmes d'optimisation globale de fonctions perturbées ou encore pour les analogies avec les systèmes de particules (machine de Bolzman).

3.1 Le principe de la rétropropagation

3.1.1 Introduction

On considère un ensemble de données ("un data set")



À partir de ce jeu de données, On aimerait pouvoir prévoir automatiquement si un point doit être dans l'équipe bleue ou l'équipe rouge.

3.1.2 Le principe

On considère un réseau de neurones $\mathcal{R}$. Il est naturellement défini par son architecture (le nombre de couches et le nombre de neurones par couche). Les fonctions d'activation et l'ensemble $P = (a_1, a_2, \dots, a_n)$ les poids de tous les neurones.
Nous pouvons associer à ce réseau une fonction $F : \mathbb{R}^n \to \mathbb{R}^p$ pour le cas où n est le nombre des entrées et p le nombre de neurones en sortie du réseau. Dans beaucoup de cas d'étude, la valeur de p sera 1.
On dispose de données correctement étiquetées (X_i, z_i) (pour $i = 1,\dots,n$) où $X_i \in \mathbb{R}^n$. Les données se décomposent :

- Une entrée de la forme $X_i = (x_{i1},\dots x_{in})$
- $z_i \in \mathbb{R}$ est la sortie attendue l'entrée X_i (ici $p = 1$).

Notre objectif est le suivant : Déterminer les poids du réseau pour que la fonction F vérifie :

$$\forall i \in 1..N, F(X_i) \cong z_i$$

Pour connaître la performance de notre modèle, il est possible de la mesurer avec par exemple cette fonction erreur :

$$E = \frac{1}{N}\sum_{i=1}^{N}(F(X_i) - z_i)^2$$

3.1.3 Descente de gradient

Vue dans le cours précédemment, Le but est de faire évoluer les poids $P = (a_1, a_2, \ldots, a_n)$ pour obtenir un meilleur réseau $\mathcal{R}$ (autrement dit la meilleure fonction F), il suffit de minimiser l'erreur E, vue comme une fonction des poids $P = (a_1, a_2, \ldots)$. Pour cela on utilise la méthode de la descente de gradient.

- On part de poids initiaux $P_0 = (a_1, a_2, \ldots)$, par exemple choisis au hasard. On fixe un pas δ.
- On construit par itérations des poids P_k selon la formule de récurrence :

$$P_{k+1} = P_k - \delta \overrightarrow{\text{grad}}\, E(P_k)$$

  À chaque itération, l'erreur $E(P_k)$ diminue. On s'arrête au bout d'un nombre d'itérations fixé à l'avance.
- Pour calculer le gradient $gradE = \sum_{i=1}^{N} gradE_i$, il faut calculer chacun des $gradE_i$, c'est-à-dire les dérivées partielles par rapport à chacun des poids a j selon la formule :

$$\frac{\partial E_i}{\partial a_j}(X_i) = 2\frac{\partial F}{\partial a_j}(X_i)(F(X_i) - z_i)$$

3.1.4 Prédiction

La conception d'un réseau de neurones est réalisée en modélisant au mieux les données injectées. Mais l'objectif réel est de faire des prédictions pour de nouvelles valeurs, jamais rencontrées auparavant. La descente de gradient produit un ensemble de poids P qui définit complètement notre réseau $\mathcal{R}$. Nous obtenons donc une fonction $F : R^n \to R^p$, construite de sorte que $F(X_i) \cong z_i$. Nous pouvons évaluer cette fonction pour tout $X \in R^n$, même pour des X différents des X_i.

3.2 Les problèmes commencent !

3.2.1 Sur-apprentissage

Le modèle obtenu « colle » parfaitement aux données d'apprentissage, mais cependant les prédictions pour de nouvelles valeurs sont mauvaises. Il s'agit donc d'un problème délicat : la fonction F obtenue vérifie bien $F(Xi) \cong zi$ pour toutes les données, mais pour une nouvelle entrée X, la sortie F(X) n'est pas une bonne prédiction. Cela se produit lorsque l'on se concentre uniquement sur l'apprentissage à partir des données, mais que l'on a oublié que le but principal est la prédiction.

3.2.2 Sous-apprentissage

Le sous-apprentissage révèle une conception correcte de l'architecture du réseau mais une mauvaise mise en œuvre. On obtient alors des poids qui ne répondent pas correctement au problème.

3.3 Universalité des réseaux de neurones

On peut voir un réseau de neurone comme une fonction. Soit :

$$F : \mathbb{R}^n \to \mathbb{R}^m$$

Où $\forall x \in \mathbb{R}^n, f(x) = \sigma_S \circ Aff_{S-1} \circ \sigma S - 1 \ldots \circ Aff_{Entre}$
Les fonctions σ sont les fonctions d'activation des couches successives. Elles peuvent être différentes parmi les nombreux choix sigmoïde, ReLu… Les fonctions Aff sont des transformations affines.
La question que l'on peut se poser est :
Étant donnée une fonction $F : \mathbb{R}^n \to \mathbb{R}^m$. Existe-t-il un réseau $\mathcal{R}$ tel que $\mathcal{R} \cong F$?
On s'intéresse donc aux sommes suivantes :

$$G(x) = \sum_{j=1}^{n} \alpha_j \sigma\left(y_i^T x + \theta_j\right)$$

Où $y_i \in \mathbb{R}^n$, $(\theta_j, \alpha_j) \in \mathbb{R}^2$ et σ est une fonction sigmoïde. Sachant qu'une fonction sigmoïde sera définie par le fait qu'elle vérifie :

$$\lim_{x\to-\infty} \sigma(x) = 0 \text{ et } \lim_{x\to+\infty} \sigma(x) = 1$$

Un exemple très simple est la fonction de Heavyside ou encore la fonction

$$\forall x \in \mathbb{R}, \sigma(x) = \frac{1}{1+e^{-x}}$$

On notera E l'ensemble des fonctions G précédente. C'est-à-dire :

$$E = \left\{ G | y_i \in \mathbb{R}^n, (\theta_j, \alpha_j) \in \mathbb{R}^2, \sigma \text{ est une fonction sigmoïde }, G(x) = \sum_{j=1}^{n} \alpha_j \sigma\left(y_i^T x + \theta_j\right) \right\}$$

Remarque 1. Les fonctions du type de G sont des fonctions caractéristiques de certains réseaux de neurones (ceux à une couche). En effet, il suffit de considérer x une entrée sur laquelle on applique un produit scalaire correspondant aux poids et aux biais de la couches de neurones. On continue par l'application de la fonction d'activation σ. La somme finale est caractérisée par les poids de sortie.

On a le résultat suivant qui est très encourageant :

Théorème 1. *(Cybenko, 1989) Soit* $C^0(I^n)$ *l'espace des fonction continues sur* $I^n = [0,1]^n$. **L'ensemble** E **est dense dans** $C^0(I^n)$.
C'est-à-dire, on a :

$$\forall f \in C^0(I^n), \forall \varepsilon > 0, \exists G \in E, \forall x \in I^n, |f(x) - G(x)| \leqslant \varepsilon$$

Remarque 2. C'est effectivement très encourageant puisque que l'on peut approcher n'importe quelle fonction continue par un réseau de neurone. **MAIS** le gros problème est que le n n'est pas quantifié ou contraint. Ce qui veut dire que le réseau qui approche le mieux une fonction (aussi simple soit-elle) peut être infiniment grand.

3.3.1 Les préliminaires théoriques

On a besoin de quelques notions théoriques avant d'aborder la preuve du thérorèmes de Cybenko.

La séparation d'ensemble Soit E un espace vectoriel normé réel et E^* l'espace dual de E. On rappelle que le dual de E est l'ensemble suivant :

$$E^* = \{\phi : E \to \mathbb{R} \text{ linéaire et continue }\}$$

Définition 2. H *un hyperplan affine de* E *est défini par :*

$$H = \{x \in E | \exists \alpha \in \mathbb{R}, \phi \in E^*, \phi(x) = \alpha\}$$

Théorème 2. *Soit* A *un compact de* E *et* B *un sous ensemble fermé de* E *(*E *de dimension finie). Si, de plus,* A *et* B *sont disjoints et connexes. Alors il existe un hyperplan* H
il existe un hyperplan H *qui sépare strictement deux ensembles* A *compact et* B *fermé non vides, disjoints et connexes.*

Théorème 3. *(de Hahn-Banach) Soient* A *et* B *deux sous ensembles de* E *et tels que* $A \cap B = \emptyset$. *Si, de plus,* A *et* B *sont convexes,* A *fermé et* B *compact. Alors il existe un hyperplan de* E *qui sépare strictement* A *et* B.

Sans la convexité, ce théorème tombe !

Remarque 3. Soit $F \in E$ un sous espace vetoriel qui n'est pas dense dans E (i.e. $\overline{F} \neq E$). ALors on a :

$$\exists \phi \in E^*, \phi \neq 0, \forall x \in F, \phi(x) = 0$$

Un peu de théorie de la mesure On propose quelques rappels utiles pour la suite.

Définition 3. *Soit* X *un ensemble et* Ω *une collection de sous-ensemble de* X. *Si on a :*
- $\emptyset \in \Omega$
- Ω *est stable par passage au complémentaire, intersection et union dénombrable.*

Alors Ω *est une tribu*

Définition 4. *On definit une mesure* μ *sur une collection d'ensemble* Ω *par :*

$$\mu : \Omega \to \mathbb{R} \cup \{+\infty\}$$

Vérifiant :
- $\mu(\emptyset) = 0$
- $\forall E \in \Omega \mu(E) \geqslant 0$
- $\mu(\cup_{i\in B} A_i) \leqslant \sum_{i\in B} \mu(A_i)$ *où B est un ensemble d'indice dénombrable*

Remarque 4.
- Une mesure est dite finie si $\forall E \in \Omega |\mu(E)| < \infty$
- une mesure μ est dite borélienne si Ω contient une tribu borélienne (c'est-à-dire la plus petite tribu qui contient tous les ouverts de E)
- μ est dite régulière si $\mu(E) = \inf\{\mu(K) | K \subset E$ mesurable et compact $\} = \sup\{\mu(A) | A \subset E$ ouvert et mesurable$\}$

Ensuite, on a besoin du résultat suivant :

Théorème 4. *(de représentation de Riesz-Markov)*
soit $\phi \in (C^0(I^n))^*$ *Alors il existe une mesure* μ *finie, borélienne et régulière telle que :*

$$\forall u \in I^n, \phi(u) = \int_{I^n} u d\mu$$

On notera $M(I^n) = \{$ensemble des mesure μ finie, borélienne et régulière sur $I^n\}$

Définition 5. $\sigma : \mathbb{R} \to \mathbb{R}$ *est dite discrimante si :*

$$\forall y \in \mathbb{R}^n, \forall \theta \in \mathbb{R}, \forall \mu \in M(I^n), \int \sigma(y^T x + \theta) d\mu(x) = 0$$

Alors μ *est identiquement nulle.*

Théorème 5. *Les fonctions sigmoïdes sont discriminantes.*

Preuve. **du théorème de Cybenko** On note :

$$S = \left\{ G(x) = \sum_{j=1}^{N} \alpha_j \sigma(y_j^T x + \theta_j) \right\}$$

S est un sous espace vectoriel de $C^0(I^n)$. C'est l'ensemble des fonctions représentative des réseaux de neurones à une couche cachée.
On va montrer par l'absurde que S est dense dans $C^0(I^n)$. On suppose donc que S n'est pas dense.
Alors on a :

$$\exists \phi \in (C^0(I^n))^*, \phi \neq 0 \text{ et } \forall u \in S, \phi(u) = 0$$

D'après le théorème de Riesz, on a l'existance de :

$$\exists \mu \in M(I^n), \forall u \in S \phi(u) = \int_{I^n} u d\mu(x)$$

En particulier, on a :

$$\forall y \in \mathbb{R}^n, \forall \theta \in \mathbb{R}, \int_{I^n} \sigma(y^T x + \theta) d\mu(x) = 0$$

Puisque σ est discriminante, alors on a $\mu = 0$. Mais alors ϕ est identiquement nulle ! C'est donc absurde. Par conséquent, $\overline{S} = C^0(I^n)$ CQFD

4 Développement des exemples