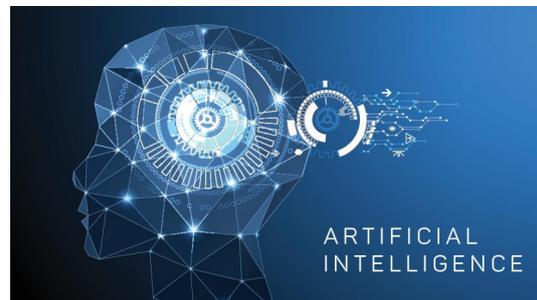


Chapitre 5 : Réseaux de neurones



Pr. Mustapha BOURAHLA, Département
d'Informatique, Université de M'Sila, Contact :
mustapha.bourahla@univ-msila.dz

Table des matières



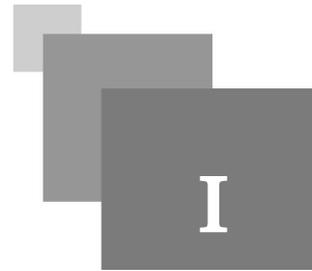
Introduction	3
I - Réseaux de neurones	4
II - Utilité des réseaux de neurones	5
III - Structure du modèle de réseaux de neurones	8
IV - Apprentissage	10
V - Différents types de réseaux de neurones	13
Conclusion	14

Introduction



Ce cinquième chapitre présente les réseaux de neurones.

Réseaux de neurones



Définition

- Un réseau de neurones artificiels ou réseau neuronal artificiel, est un système dont la conception est à l'origine schématiquement inspirée du fonctionnement des neurones biologiques, et qui par la suite s'est rapproché des méthodes statistiques.
- Les réseaux de neurones sont généralement optimisés par des méthodes d'apprentissage de type probabiliste, en particulier bayésien.
- Ils sont placés d'une part dans la famille des applications statistiques, qu'ils enrichissent avec un ensemble de paradigmes permettant de créer des classifications rapides (réseaux de Kohonen en particulier), et d'autre part dans la famille des méthodes de l'intelligence artificielle auxquelles ils fournissent un mécanisme perceptif indépendant des idées propres de l'implémenteur, et des informations d'entrée au raisonnement logique formel (Deep Learning).
- En modélisation des circuits biologiques, ils permettent de tester quelques hypothèses fonctionnelles issues de la neurophysiologie, ou encore les conséquences de ces hypothèses pour les comparer au réel.

Utilité des réseaux de neurones

II

Utilité

- Les réseaux de neurones, en tant que systèmes capables d'apprendre, mettent en œuvre le principe de l'induction, c'est-à-dire l'apprentissage par l'expérience.
- Par confrontation avec des situations ponctuelles, ils infèrent un système de décision intégré dont le caractère générique est fonction du nombre de cas d'apprentissages rencontrés et de leur complexité par rapport à la complexité du problème à résoudre.
- Par opposition, les systèmes symboliques capables d'apprentissage, s'ils implémentent également l'induction, le font sur base de la logique algorithmique, par complexification d'un ensemble de règles déductives (Prolog par exemple).
- Grâce à leur capacité de classification et de généralisation, les réseaux de neurones sont généralement utilisés dans des problèmes de nature statistique, tels que la classification automatique de codes postaux ou la prise de décision concernant un achat boursier en fonction de l'évolution des cours.
- Autre exemple, une banque peut créer un jeu de données sur les clients qui ont effectué un emprunt constitué : de leur revenu, de leur âge, du nombre d'enfants à charge... et s'il s'agit d'un bon client. Si ce jeu de données est suffisamment grand, il peut être utilisé pour l'entraînement d'un réseau de neurones.
- La banque pourra alors présenter les caractéristiques d'un potentiel nouveau client, et le réseau répondra s'il sera bon client ou non, en généralisant à partir des cas qu'il connaît.
- Si le réseau de neurones fonctionne avec des nombres réels, la réponse traduit une probabilité de certitude. Par exemple : 1 pour « sûr qu'il sera un bon client », -1 pour « sûr qu'il sera mauvais client », 0 pour « aucune idée », 0,9 pour « presque sûr qu'il sera bon client ».
- Le réseau de neurones ne fournit pas toujours de règle exploitable par un humain. Le réseau reste souvent une boîte noire qui fournit une réponse quand on lui présente une donnée, mais le réseau ne fournit pas de justification facile à interpréter.

Les réseaux de neurones sont réellement utilisés, par exemple :

- pour la classification d'espèces animales par espèce étant donnée une analyse ADN.
- reconnaissance de motif ; par exemple pour la reconnaissance optique de caractères (OCR), et notamment par les banques pour vérifier le montant des chèques, par La Poste pour trier le courrier en fonction du code postal, etc. ; ou bien encore pour le déplacement automatisé de robots mobiles autonomes.
- approximation d'une fonction inconnue.
- modélisation accélérée d'une fonction connue mais très complexe à calculer avec exactitude ; par exemple certaines fonctions d'inversions utilisées pour décoder les signaux de télé-détection émis par les satellites et les transformer en données sur la surface de la mer.

- estimations boursières :
 - apprentissage de la valeur d'une entreprise en fonction des indices disponibles : bénéfices, endettements à long et court terme, chiffre d'affaires, carnet de commandes, indications techniques de conjoncture. Ce type d'application ne pose pas en général de problème
 - tentatives de prédiction sur la périodicité des cours boursiers. Ce type de prédiction est très contesté pour deux raisons, l'une étant qu'il n'est pas évident que le cours d'une action ait de façon tout à fait convaincante un caractère périodique (le marché anticipe en effet largement les hausses comme les baisses prévisibles, ce qui applique à toute périodicité éventuelle une variation de période tendant à la rendre difficilement fiable), et l'autre que l'avenir prévisible d'une entreprise détermine au moins aussi fortement le cours de son action, si ce n'est plus encore que peut le faire son passé ; les cas de Pan Am, Manufrance ou IBM permettent de s'en convaincre.
- modélisation de l'apprentissage et amélioration des techniques de l'enseignement.
- en météorologie, pour la classification de conditions atmosphériques et la prévision statistique du temps.
- en auscultation des ouvrages hydrauliques, pour la compréhension physique des phénomènes de déplacements, sous-pressions et débits de fuite.

Limites

- Les réseaux de neurones artificiels ont besoin de cas réels servant d'exemples pour leur apprentissage (on appelle cela la base d'apprentissage).
- Ces cas doivent être d'autant plus nombreux que le problème est complexe et que sa topologie est peu structurée.
- Ainsi on peut optimiser un système neuronal de lecture de caractères en utilisant le découpage manuel d'un grand nombre de mots écrits à la main par de nombreuses personnes.
- Chaque caractère peut alors être présenté sous la forme d'une image brute, disposant d'une topologie spatiale à deux dimensions, ou d'une suite de segments presque tous liés. La topologie retenue, la complexité du phénomène modélisé, et le nombre d'exemples doivent être en rapport.
- Sur un plan pratique, cela n'est pas toujours facile car les exemples peuvent être soit en quantité absolument limitée ou trop onéreux à collecter en nombre suffisant.
- Il y a des problèmes qui se traitent bien avec les réseaux de neurones, en particulier ceux de classification en domaines convexes (c'est-à-dire tels que si des points A et B font partie du domaine, alors tout le segment AB en fait partie aussi).
- Des problèmes comme « Le nombre d'entrées à 1 (ou à zéro) est-il pair ou impair ? » se résolvent en revanche très mal : pour affirmer de telles choses sur 2 puissance N points, si on se contente d'une approche naïve mais homogène, il faut précisément N-1 couches de neurones intermédiaires, ce qui nuit à la généralité du procédé.
- Un exemple caricatural, mais significatif est le suivant : disposant en entrée du seul poids d'une personne, le réseau doit déterminer si cette personne est une femme ou bien un homme. Les femmes étant statistiquement un peu plus légères que les hommes, le réseau fera toujours un peu mieux qu'un simple tirage au hasard : cet exemple dépouillé indique la simplicité et les limitations de ces modèles mais il montre également comment l'étendre : l'information « port d'une jupe », si on l'ajoute, aurait clairement un coefficient synaptique plus grand que la simple information de poids.

Opacité

- Les réseaux complexes de neurones artificiels ne peuvent généralement pas expliquer eux-mêmes leur façon de « penser ».
- Les calculs aboutissant à un résultat ne sont pas visibles pour les programmeurs qui ont créé le réseau neuronal.
- Une « neuroscience de l'intelligence artificielle » a donc été créée pour étudier la boîte noire que constituent les réseaux de neurones, science qui pourrait permettre d'augmenter la confiance dans les résultats produits par ces réseaux ou les intelligences artificielles qui les utilisent.

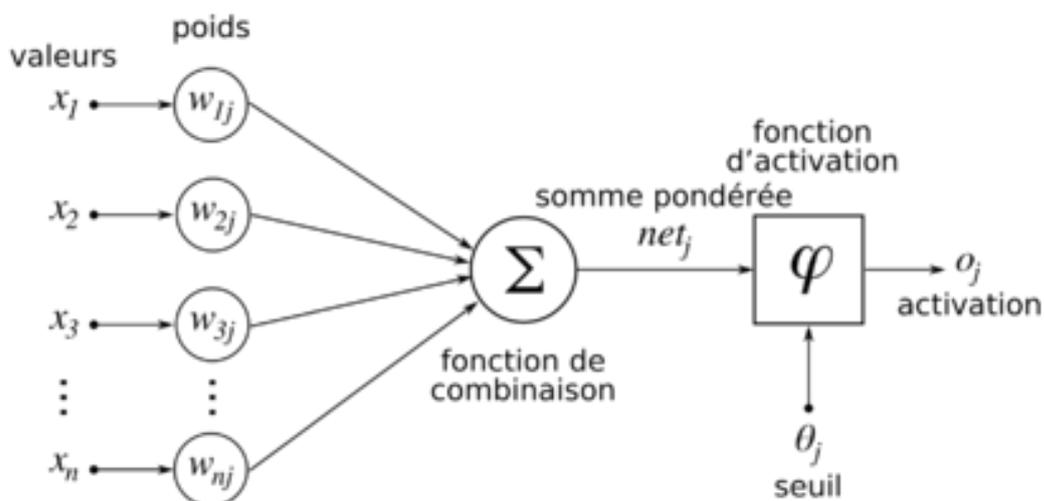
Structure du modèle de réseaux de neurones

III

Modèle

Structure du réseau

- Un réseau de neurones est en général composé d'une succession de couches dont chacune prend ses entrées sur les sorties de la précédente.
- Chaque couche (i) est composée de N_i neurones, prenant leurs entrées sur les N_{i-1} neurones de la couche précédente.
- À chaque synapse est associé un poids synaptique, de sorte que les N_{i-1} sont multipliés par ce poids, puis additionnés par les neurones de niveau i , ce qui est équivalent à multiplier le vecteur d'entrée par une matrice de transformation.
- Mettre l'une derrière l'autre les différentes couches d'un réseau de neurones reviendrait à mettre en cascade plusieurs matrices de transformation et pourrait se ramener à une seule matrice, produit des autres, s'il n'y avait à chaque couche, la fonction de sortie qui introduit une non linéarité à chaque étape.
- Ceci montre l'importance du choix judicieux d'une bonne fonction de sortie : un réseau de neurones dont les sorties seraient linéaires n'aurait aucun intérêt.
- Au-delà de cette structure simple, le réseau de neurones peut également contenir des boucles qui en changent radicalement les possibilités mais aussi la complexité.
- De la même façon que des boucles peuvent transformer une logique combinatoire en logique séquentielle les boucles dans un réseau de neurones transforment un simple dispositif de reconnaissance d'entrées en une machine complexe capable de toutes sortes de comportements.



Structure d'un neurone artificiel ou neurone formel. Le neurone calcule la somme de ses entrées x , pondérée par les poids w , puis cette valeur passe à travers la fonction d'activation ϕ pour produire sa sortie o .

Fonction de combinaison

Considérons un neurone quelconque.

Il reçoit des neurones en amont un certain nombre de valeurs via ses connexions synaptiques, et il produit une certaine valeur en utilisant une fonction de combinaison. Cette fonction peut donc être formalisée comme étant une fonction vecteur-à-scalaire, notamment :

- Les réseaux de type MLP (multi-layer perceptron) calculent une combinaison linéaire des entrées, c'est-à-dire que la fonction de combinaison renvoie le produit scalaire entre le vecteur des entrées et le vecteur des poids synaptiques.
- Les réseaux de type RBF (radial basis function) calculent la distance entre les entrées, c'est-à-dire que la fonction de combinaison renvoie la norme euclidienne du vecteur issu de la différence vectorielle entre les vecteurs d'entrées.

Fonction d'activation

La fonction d'activation (ou fonction de seuillage, ou encore fonction de transfert) sert à introduire une non-linéarité dans le fonctionnement du neurone.

Les fonctions de seuillage présentent généralement trois intervalles :

1. en dessous du seuil, le neurone est non-actif (souvent dans ce cas, sa sortie vaut 0 ou -1) ;
2. aux alentours du seuil, une phase de transition ;
3. au-dessus du seuil, le neurone est actif (souvent dans ce cas, sa sortie vaut 1).

Des exemples classiques de fonctions d'activation sont :

1. La fonction sigmoïde.
2. La fonction tangente hyperbolique.
3. La fonction de Heaviside.

La logique bayésienne, dont le théorème de Cox-Jaynes formalise les questions d'apprentissage, fait intervenir

aussi une fonction en S qui revient de façon récurrente : $ev(p) = 10 \log \left(\frac{p}{1-p} \right)$

Propagation de l'information

- Ce calcul effectué, le neurone propage son nouvel état interne sur son axone.
- Dans un modèle simple, la fonction neuronale est simplement une fonction de seuillage : elle vaut 1 si la somme pondérée dépasse un certain seuil ; 0 sinon.
- Dans un modèle plus riche, le neurone fonctionne avec des nombres réels (souvent compris dans l'intervalle $[0,1]$ ou $[-1,1]$). On dit que le réseau de neurones passe d'un état à un autre lorsque tous ses neurones recalculent en parallèle leur état interne, en fonction de leurs entrées.

Apprentissage

IV

Apprentissage

Base théorique

- La notion d'apprentissage, bien que connue déjà, n'est pas modélisable dans le cadre de la logique déductive : celle-ci en effet procède à partir de connaissances déjà établies dont on tire des connaissances dérivées. Or il s'agit ici de la démarche inverse : par observations limitées, tirer des généralisations plausibles : c'est un procédé par induction.
- La notion d'apprentissage recouvre deux réalités souvent traitées de façon successive :
 1. mémorisation : le fait d'assimiler sous une forme dense des exemples éventuellement nombreux,
 2. généralisation : le fait d'être capable, grâce aux exemples appris, de traiter des exemples distincts, encore non rencontrés, mais similaires.
- Dans le cas des systèmes d'apprentissage statistique, utilisés pour optimiser les modèles statistiques classiques, réseaux de neurones et automates markoviens, c'est la généralisation qui est l'objet de toute l'attention.
- Cette notion de généralisation est traitée de façon plus ou moins complète par plusieurs approches théoriques.
 1. La généralisation est traitée de façon globale et générique par la théorie de la régularisation statistique introduite par Vladimir Vapnik.
 Cette théorie, développée à l'origine en Union soviétique, s'est diffusée en Occident depuis la chute du mur de Berlin.
 La théorie de la régularisation statistique s'est diffusée très largement parmi ceux qui étudient les réseaux de neurones en raison de la forme générique des courbes d'erreurs résiduelles d'apprentissage et de généralisation issues des procédures d'apprentissage itératives telles que les descentes de gradient utilisées pour l'optimisation des perceptrons multi-couches.
 Ces formes génériques correspondent aux formes prévues par la théorie de la régularisation statistique ; cela vient du fait que les procédures d'apprentissage par descente de gradient, partant d'une configuration initiale des poids synaptiques explorent progressivement l'espace des poids synaptiques possibles ; on retrouve alors la problématique de l'augmentation progressive de la capacité d'apprentissage, concept fondamental au cœur de la théorie de la régularisation statistique.
 2. La généralisation est aussi au cœur de l'approche de l'inférence bayésienne, enseignée depuis plus longtemps.
 Le théorème de Cox-Jaynes fournit ainsi une base importante à un tel apprentissage, en nous apprenant que toute méthode d'apprentissage est soit isomorphe aux probabilités munies de la relation de Bayes, soit incohérente. C'est là un résultat extrêmement fort, et c'est pourquoi les méthodes bayésiennes sont largement utilisées dans le domaine.

Classe de problèmes solubles

- En fonction de la structure du réseau, différents types de fonction sont approchables grâce aux réseaux de neurones :
- Fonctions représentables par un perceptron
- Un perceptron (un réseau à une unité) peut représenter les fonctions booléennes suivantes : and, or, nand, nor mais pas le xor.

Comme toute fonction booléenne est représentable à l'aide de ces fonctions, un réseau de perceptrons est capable de représenter toutes les fonctions booléennes.

En effet les fonctions nand et nor sont dites universelles : on peut par combinaison de l'une de ces fonctions représenter toutes les autres.

- Fonctions représentables par des réseaux de neurones multicouches acycliques
 1. Fonctions booléennes : toutes les fonctions booléennes sont représentables par un réseau à deux couches. Au pire des cas, le nombre de neurones de la couche cachée augmente de manière exponentielle en fonction du nombre d'entrées.
 2. Fonctions continues : toutes les fonctions continues bornées sont représentables, avec une précision arbitraire, par un réseau à deux couches (Cybenko, 1989).
Ce théorème s'applique au réseau dont les neurones utilisent la sigmoïde dans la couche cachée et des neurones linéaires (sans seuil) dans la couche de sortie.
Le nombre de neurones dans la couche cachée dépend de la fonction à approximer.
 3. Fonctions arbitraires : n'importe quelle fonction peut être approximée avec une précision arbitraire grâce à un réseau à trois couches (théorème de Cybenko, 1988).

Algorithme

- La large majorité des réseaux de neurones possède un algorithme «d'entraînement» qui consiste à modifier les poids synaptiques en fonction d'un jeu de données présentées en entrée du réseau.
- Le but de cet entraînement est de permettre au réseau de neurones d'«apprendre» à partir des exemples.
- Si l'entraînement est correctement réalisé, le réseau est capable de fournir des réponses en sortie très proches des valeurs d'origine du jeu de données d'entraînement.
- Mais tout l'intérêt des réseaux de neurones réside dans leur capacité à généraliser à partir du jeu de test.
- Il est donc possible d'utiliser un réseau de neurones pour réaliser une mémoire ; on parle alors de mémoire neuronale.
- La vision topologique d'un apprentissage correspond à la détermination de l'hypersurface sur \mathbb{R}^n où \mathbb{R} est l'ensemble des réels, et n le nombre d'entrées du réseau.

Apprentissage

Mode supervisé ou non

- Un apprentissage est dit supervisé lorsque le réseau est forcé à converger vers un état final précis, en même temps qu'un motif lui est présenté.
- À l'inverse, lors d'un apprentissage non-supervisé, le réseau est laissé libre de converger vers n'importe quel état final lorsqu'un motif lui est présenté.

Surapprentissage

- Il arrive souvent que les exemples de la base d'apprentissage comportent des valeurs approximatives ou bruitées. Si on oblige le réseau à répondre de façon quasi parfaite relativement à ces exemples, on peut obtenir un réseau qui est biaisé par des valeurs erronées.
- Par exemple, imaginons qu'on présente au réseau des couples $(x_i, f(x_i))$ situés sur une droite d'équation $y = ax + b$, mais bruités de sorte que les points ne soient pas exactement sur la droite.
- S'il y a un bon apprentissage, le réseau répond $ax + b$ pour toute valeur de x présentée.
- S'il y a du surapprentissage, le réseau répond un peu plus que $ax + b$ ou un peu moins, car chaque couple $(x_i, f(x_i))$ positionné en dehors de la droite va influencer la décision : il aura appris le bruit en plus, ce qui n'est pas souhaitable.
- Pour éviter le surapprentissage, il existe une méthode simple : il suffit de partager la base d'exemples en 2 sous-ensembles. Le premier sert à l'apprentissage et le second sert à l'évaluation de l'apprentissage. Tant que l'erreur obtenue sur le deuxième ensemble diminue, on peut continuer l'apprentissage, sinon on arrête.

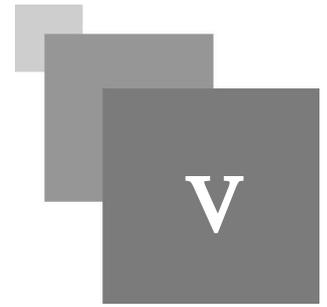
Rétropropagation

- La rétropropagation consiste à rétropropager l'erreur commise par un neurone à ses synapses et aux neurones qui y sont reliés.
- Pour les réseaux de neurones, on utilise habituellement la rétropropagation du gradient de l'erreur, qui consiste à corriger les erreurs selon l'importance des éléments qui ont justement participé à la réalisation de ces erreurs : les poids synaptiques qui contribuent à engendrer une erreur importante se verront modifiés de manière plus significative que les poids qui ont engendré une erreur marginale.

Élagage

- L'élagage (pruning, en anglais) est une méthode qui permet d'éviter le surapprentissage tout en limitant la complexité du modèle.
- Elle consiste à supprimer des connexions (ou synapses), des entrées ou des neurones du réseau une fois l'apprentissage terminé.
- En pratique, les éléments qui ont la plus petite influence sur l'erreur de sortie du réseau sont supprimés.
- Deux exemples d'algorithmes d'élagage sont :
 1. Optimal brain damage (OBD) de Yann LeCun et al.
 2. Optimal brain surgeon (OBS) de B. Hassibi et D. G. Stork

Différents types de réseaux de neurones



Différents types de réseaux de neurones

- L'ensemble des poids des liaisons synaptiques détermine le fonctionnement du réseau de neurones.
- Les motifs sont présentés à un sous-ensemble du réseau de neurones : la couche d'entrée. Lorsqu'un motif est appliqué à un réseau, celui-ci cherche à atteindre un état stable.
- Lorsqu'il est atteint, les valeurs d'activation des neurones de sortie constituent le résultat.
- Les neurones qui ne font ni partie de la couche d'entrée ni de la couche de sortie sont dits neurones cachés.
- Les types de réseau de neurones diffèrent par plusieurs paramètres :
 1. la topologie des connexions entre les neurones ;
 2. la fonction d'agrégation utilisée (somme pondérée, distance pseudo-euclidienne...);
 3. la fonction de seuillage utilisée (sigmoïde, échelon, fonction linéaire, fonction de Gauss...);
 4. l'algorithme d'apprentissage (rétropropagation du gradient, cascade corrélation);
 5. d'autres paramètres, spécifiques à certains types de réseaux de neurones, tels que la méthode de relaxation pour les réseaux de neurones (réseaux de Hopfield par exemple) qui ne sont pas à propagation simple (perceptron multicouche par exemple).
- De nombreux autres paramètres sont susceptibles d'être mis en œuvre dans le cadre de l'apprentissage de ces réseaux de neurones par exemple :
 - la méthode de dégradation des pondérations (weight decay), permettant d'éviter les effets de bord et de neutraliser le surapprentissage.

Conclusion



Un aperçu sur les réseaux de neurones est présenté dans ce chapitre.