

---

---

# Automatisation du diagnostic de la pathologie cognitive TDA/H sur base d'acquisition EEG via le deep learning

---

---

\*\*\*\*\*

---

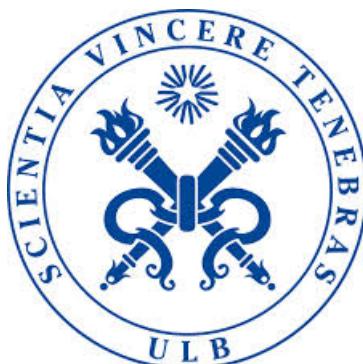
---

Ecrit par :

DANY S EFILA

Sous la supervision de :

TOM LENAERTS, YANN-AËL LE BORGNE, NASSIM VERSBRAEGEN



Département des sciences informatiques  
UNIVERSITÉ LIBRE DE BRUXELLES

Une thèse portant sur un sujet scientifique valide par l'institution est défendu dans le but d'obtenir le diplôme de MASTER EN SCIENCES INFORMATIQUE de la faculté des sciences.

MAI 2018



## INTRODUCTION

### 1.1 Introduction

Les recherches et les avancées technologiques, de l'ingénierie informatique, des neurosciences ainsi que des sciences informatiques ont permis la convergence de domaines d'activités différents. Le Trouble du Déficit de l'Attention avec Hyperactivité (TDA/H) est une pathologie cognitive dont le diagnostic médical nécessite une batterie de tests et d'examens [11]. Ce processus étant très lourd, de nombreuses recherches et expérimentations ont été faites, pour faciliter le diagnostic en se basant sur des techniques utilisant des réseaux de neurones. Les applications des réseaux de neurones sont multiples. Par exemple, on a développé un auto-pilote pour avion, ou encore un système de guidage pour automobile, on a conçu des systèmes de lecture automatique de chèques bancaires et d'adresses postales, on produit des systèmes de traitement du signal pour différentes applications militaires, un système pour la synthèse de la parole, des réseaux sont aussi utilisés pour bâtir des systèmes de vision par ordinateur, pour faire des prévisions sur les marchés monétaires, pour évaluer le risque financier ou en assurance, pour différents processus manufacturiers, pour le diagnostic médical, pour l'exploration pétrolière ou gazière, en robotique, en télécommunication, et bien d'autres domaine encore [14]. Le projet TDA/H de Human Waves a pour but de pouvoir utiliser des réseaux de neurones pour automatiser le diagnostic de la TDA/H en se basant sur des relevés d'électroencéphalographies de patients. Mon travail portera sur la conception et la réalisation de ce projet. Ce chapitre introductif fixe le contexte du sujet, présente les objectifs, et définit la structure du document.

### 1.1.1 Contexte du projet TAD/H

Des entreprises telles que Human Waves développent des produits et proposent des services issus des neurosciences et présentant des applications très diverses. Human Waves est une spin-off du Laboratoire de neurophysiologie et de biomécanique du mouvement de la faculté des sciences de l'Université Libre de Bruxelles. Ses activités se divisent en trois grandes catégories : «human performance», «human health», et «human engineering». Dans la section «human health» l'entreprise conçoit des applications dans le domaine médical, visant à l'aide au diagnostic de pathologie chez les jeunes enfants tel que le trouble de l'attention avec ou sans hyper activité (TDA/H) ainsi que des méthodes de traitement de cette maladie. Pour ce faire, les outils de diagnostic utilisés sont basés sur l'analyse d'électroencéphalographies des sujets soumis à des stimuli (réalisation d'une tâche cognitive) suivant le protocole GO/NO GO dont nous parlerons plus en détails dans la suite du chapitre. Les recherches dans ledit domaine s'orientent vers deux aspects majeurs [8].

- L'identification d'un marqueur biologique précoce non invasif et fiable

– La mise en place d'une méthode de traitement non médicamenteux

Le travail sur lequel porte ce document vise à apporter une solution au premier aspect.

### 1.1.2 Objectifs

Le diagnostic de la pathologie TDAH dépend de certains critères qui sont connus, comme l'analyse des bandes de fréquences renseignées sur un EEG, l'amplitude et la latence des ERPs. Toutefois des facteurs extérieurs peuvent influer ces mesures et fausser le résultat. Des procédés liés à l'usage des méthodes d'apprentissage automatique et des classificateurs ont déjà fait l'objet de nombreux articles tels que la classification des EEG pour les interfaces cerveau machine [15], la modélisation des ondes EEG via un réseau deep belief semi supervisé [21]. Ces procédés sont d'autant plus fiables car ils apprennent de manière autonome et sont capables de prendre en compte un plus grand nombre de critères évaluatifs. Human Waves développe un projet lié au diagnostic et au traitement du TDAH (le projet neuroAt), le but des recherches actuelles est de pouvoir prendre en compte toutes les informations physiologiques ressortant d'un EEG et de statuer sur leur pertinence ou non dans l'élaboration du diagnostic. L'idée est de faire un nombre suffisant d'enregistrement (environ une centaine) d'EEG, d'en extraire le contenu prenant ainsi le maximum d'informations et de le soumettre à un processus capable de discerner les enfants TDAH et non TDAH. Il en ressort donc que le principal but serait d'appliquer un procédé intelligent (deep learning) pour faire une classification d'individus selon qu'ils soient TDAH ou non.

**Buts visés par le travail** La méthode actuelle n'est pas totalement fiable car elle se fonde sur des analyses manuelles et des analyses statistiques basées sur les statistiques de permutations. Ceci s'explique par le fait que l'analyse des EEG se fait d'abord manuellement par un neuropsychologue (donc sujet à des erreurs d'interprétations potentielles) puis les ERPs sont évalués en réponse à une excitation cognitive. Les courbes de moyenne des ERP (moyenne de puissance des ondes cérébrales) que l'on obtient sur un échantillon donné, peuvent différer après une nouvelle application de statistiques. Ceci est dû d'une part au degré de précision des statistiques, et d'autre part au fait que les données analysées sont des données physiologiques et qu'elles diffèrent d'un sujet à l'autre. Le but est donc de pouvoir proposer une solution plus fiable qui serait basée sur l'implémentation d'un réseau de neurones en remplacement des étapes de traitements des EEG et des statistiques, qui prendrait en entrées des EEG et qui après plusieurs traitements au sein du réseau parviendrait à classifier l'appartenance du sujet au groupe de TDAH ou à celui de contrôles.

### 1.1.3 Structure du document

Le document est organisé comme suit. Nous présentons premièrement dans la section de l'état de l'art, les outils théoriques du deep learning nécessaires, les applications du deep learning dans la classification de la TDA/H; puis dans la section suivante les contributions envisagées. Dans cette partie il sera question de proposer une modélisation de solution, de suggérer des outils et programmes pour implémenter la solution, puis les résultats obtenus après l'implémentation et enfin une discussion suivie d'une conclusion générale sur l'ensemble du travail.

Selon la définition des besoins de notre problème, il convient d'appliquer le deep learning sur un ensemble de données issues des relevés d'électroencéphalographies. C'est pourquoi un passage en revue, du deep learning, de son rôle, de son fonctionnement, de ses applications, et de ses types d'architectures est indispensable. Le trouble de l'attention étant une pathologie cérébrale, l'électroencéphalographie s'avère être un moyen efficace pour l'analyser.

## 1.2 Fonctionnement électrique du cerveau

L'origine de l'activité électrique enregistrable du cerveau se situe dans les cellules nerveuses, les neurones, qui composent cet organe. Un neurone est composé de différentes parties : les dendrites, le corps cellulaire, l'axone et les synapses [16].

La tension électrique dans notre cerveau est créée suivant un processus chimique. L'excitation d'un neurone à travers une synapse entraîne l'ouverture de canaux ioniques au niveau de sa membrane. La composition ionique étant différente à l'intérieur et à l'extérieur des cellules, l'ouverture des canaux engendre un mouvement de particules chargées dans le milieu intra- et

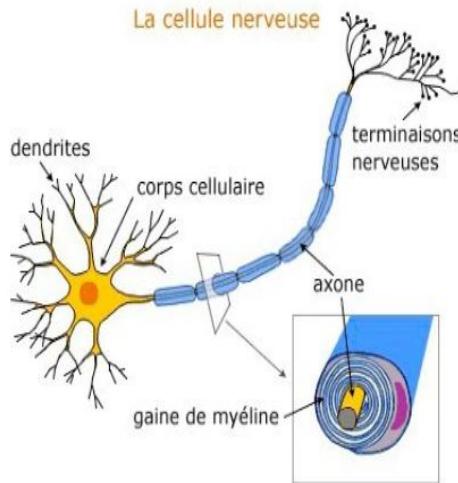


FIGURE 1.1 – schéma d'un neurone biologique [1]

extracellulaire. Ces courants dits ‘primaires’ sont à l’origine des mesures EEG . Ce sont les courants post-synaptiques engendrés dans les dendrites des cellules qui sont captés majoritairement sur le scalp [7]. Les courants électriques qui circulent via nos neurones sont à l’origine de la pensée, de la locution, de la motricité, de la vue et d’autres aptitudes dont est doté l’être humain.

### 1.2.1 Les types d’ondes électriques mesurées

Le cerveau a une activité électrique, qui peut, comme toute activité électrique, être mesurée par un oscilloscope. Dans le cas de l’activité cérébrale, on utilise des électroencéphalogrammes. Le cerveau possède 5 principaux rythmes dont les bandes de fréquences sont les suivantes [11] :

1. Les ondes delta (0. 1 - 4 Hz);
2. Les ondes thêta (4 - 7 Hz);
3. Les ondes alpha (8 - 12 Hz);
4. Les ondes bêta (12 - 40 Hz);
5. Les ondes gamma (40 - 80 Hz);

La figure ci dessous représente l’allure des ondes dans chacune des bandes fréquentielles.

L’analyse de la puissance moyenne de chacune de ces ondes cérébrales via un EEG pendant la durée d’un stimulus peut nous permettre de déceler des pathologies cérébrales.

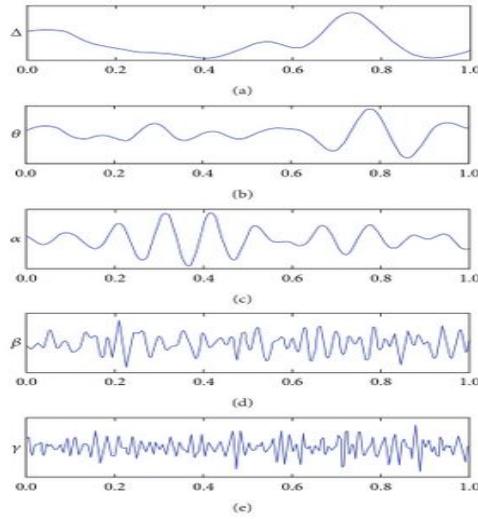


FIGURE 1.2 – rythmes cérébraux,source [9]

### 1.3 Électroencéphalographie

Elle consiste à enregistrer directement l'activité électrique du cerveau à partir d'électrodes placées à la surface du scalp. Grâce à leur excellente résolution temporelle, de l'ordre de la milliseconde, elles permettent de suivre en temps réel la chronologie des opérations mentales, et d'étudier la dynamique des phénomènes cérébraux [7]. L'EEG permet surtout de mesurer la différence de potentiel entre les électrodes réparties régulièrement sur le scalp. Le nombre d'électrodes utilisées peut être très variable, allant de 20 électrodes utilisées pour la clinique à des nombres plus importants 64, 128 voire 256 essentiellement utilisés pour les Sciences Cognitives [7]. Ces électrodes sont disposées dans un bonnet électrique que l'on pose sur la tête du patient le temps de l'enregistrement. L'EEG a beaucoup été utilisé comme un outil d'interprétation et d'aide au diagnostic par exemple en repérant et classifiant les pointes épileptiques ou en détectant des anomalies d'activités rythmiques cérébrales [7].

L'application aux neurosciences cognitives nécessitait de faire ressortir de l'activité spontanée du cerveau celle liée à la perception d'un stimulus, et au traitement d'informations requis pour accomplir une tâche. En sommant des tracés issus de la répétition de la même tâche, il est possible de faire apparaître des composantes spécifiques, appelées réponses évoquées [7].

En fonction des besoins les EEG peuvent être analysés dans le domaine temporel ou fréquentiel.

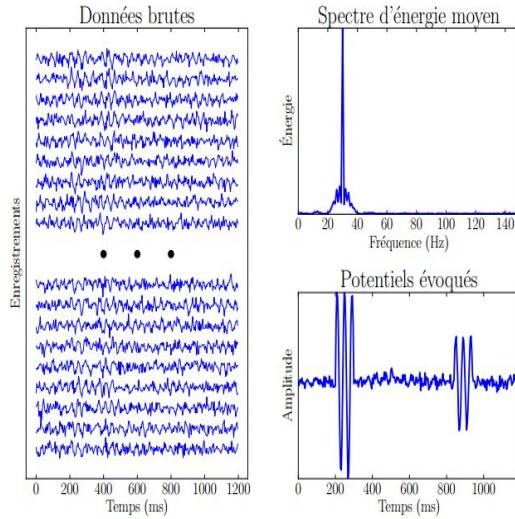


FIGURE 1.3 – Spectre d'EEG, source [16]

#### 1.3.0.1 Analyse fréquentielle des EEG

Le diagnostic des troubles cognitifs a été l'une des premières applications de l'analyse spectrale (analyse fréquentielle) des EEG [5]. Les signaux du cortex cérébral se présentent sous une forme oscillatoire, pour déterminer leur fréquence d'oscillation l'une des méthodes est la FFT (Fast Fourier Transformation). Il s'agit d'une décomposition du signal comme une somme pondérée de sinusoïdes complexes. Les coefficients étant complexes, il est possible de calculer une amplitude et une phase pour chaque sinusoïde. L'information d'amplitude peut alors être utilisée pour illustrer le contenu fréquentiel du signal dans un spectre d'énergie. Le signal de base étant bruité, le spectre d'énergie est en général obtenu en moyennant les spectres des signaux de plusieurs enregistrements d'une même expérience [16].

#### 1.3.0.2 Analyse temporelle des EEG

Dans le domaine temporel, on évalue surtout les réponses cérébrales d'un individu à un stimulus pendant la durée du stimulus. Le but est de déceler dans le signal des événements liés à la présentation du stimulus ou à la réponse par le sujet [17]. En pratique, un grand nombre d'enregistrements, supérieur à une centaine (dans notre cas 5 x 60 enregistrements), d'une même expérience est réalisé. Puis, le signal de tous les enregistrements est aligné sur le temps du stimulus ou de la réponse avant de calculer une moyenne en chaque point temporel sur l'ensemble des enregistrements. Le signal moyen obtenu est dit « verrouillé » sur le stimulus ou sur la réponse [16]. Il est possible qu'on voit apparaître dans le signal moyen de larges déviations du potentiel enregistré, ces dernières représentent des pointes d'ondes [16], il s'agit des potentiels évoqués (Event Related Potential, ERP). Il faut toutefois noter que cette méthode ne permet pas de prendre en compte les potentiels induits lors de la quantification du signal [17].

### 1.3.0.3 Analyse temps - fréquence : méthode ERSP

#### Analyse temps - fréquence : méthode ERSP



**Event Related Spectral Perturbations (ERSP)** Est une méthode permettant de combiner les deux manières d'analyser les signaux cérébraux précédentes, et de palier à leur limitation . Une méthode d'analyse des ERSP se fonde sur une transformée temps-fréquence (exemple de la transformée de Fourier à court terme) pour calculer dans un premier temps la densité d'énergie temps-fréquence du signal de chaque enregistrement (le but est d'obtenir un spectre d'énergie fréquentiel en chaque point temporel du signal). La seconde étape d'une méthode d'analyse ERSP est le calcul d'une moyenne des spectres d'énergie temps-fréquence sur l'ensemble des enregistrements, en verrouillant sur le stimulus ou sur la réponse à la manière de l'analyse des potentiels évoqués [17].



**Autres méthodes** Une autre manière d'interpréter les EEG est l'analyse non linéaire. Depuis les années 90, ces méthodes ont été utilisées en neurophysiologie, en épileptologie et pour le diagnostic ainsi que le traitement de la maladie d'Alzheimer [5].

L'utilisation des EEG dans l'aide au diagnostic se fait de plus en plus. Analyser avec l'une ou l'autre des méthodes, on peut les utiliser dans le cadre de détection de troubles ou de pathologie comme l'épilepsie, l'autisme, le retard mental, troubles spécifiques du langage oral, troubles envahissants du développement [4]. Malgré sa grande utilité il n'est pas le seul moyen susceptible de donner des informations liées à des troubles cérébraux, les MEG (Magnéto encéphalographie), les IRM f (Imagerie par Résonance Magnétique fonctionnelle) et les TEP (Tomographie par émission de positons) [6] dont nous ne parlerons pas plus en détails ont également le même objectif.



## 1.4 La pathologie TDA/H ou TDAH

Le Trouble du Déficit de l'Attention avec Hyperactivité est un trouble qui touche avec une prévalence de 3 % à 5% les jeunes enfants âgés entre 8 et 12 ans [12] et dont les symptômes peuvent perdurer jusqu'à l'âge adulte. Il est caractérisé par un niveau élevé d'inattention et/ou d'hyperactivité ainsi qu'une grande impulsivité [12] Plusieurs observations permettent d'en faire le diagnostic, notamment celle des marqueurs EEG en plus d'autres symptômes liés au comportement des sujets dans les situations de la vie courante (mémorisation, locution, concentration. . . ) [11], [5].

### 1.4.1 L'EEG dans le cas de la détection de l'TDAH

Comme nous l'avons présenté dans la section précédente, les ondes cérébrales sont divisées en cinq bandes de fréquences relevées via un EEG. Les ERPs (issus d'une analyse temporelle des EEG) au travers de leur amplitude ou de leur latence suivant le protocole de neurofeedback aident à faire une classification entre les sujets TDAH et non TDAH. **L'utilisation de ce critère est précis à 73% chez les enfants et à 59% chez les adolescents** [10]. Les ondes neuronales intervenant dans la pathologie TDAH sont les ondes Thêta, Alpha et Bêta, elle peuvent être observées individuellement ou en relation les unes par rapport aux autres. ces observations se font dans différents états : au repos, avec les yeux ouverts, avec les yeux fermés, en plein travail de mémorisation . Les allures des ondes vont varier et suivant les valeurs du ratio Thêta / bêta mesurées on en déduit le diagnostic TDAH. Néanmoins cette approche n'est pas totalement fiable, car d'autres facteurs comme l'âge des personnes et des facteurs neurologiques peuvent influencer ces ondes [11].

**Diagnostic basé sur le rapport de puissance thêta/bêta** Des études antérieures ont examinées la validité discriminante du rapport thêta/bêta dans la détection de l'TDAH. Dans **leurs** méta-analyses, trouvèrent un critère permettant de prédire une sensibilité et une spécificité de 94% [11]. Une étude pilote réalisée sur 159 enfants et adolescents suspectés d'avoir l'TDAH avec un taux de 61% diagnostiqués TDAH, qui devaient se rendre dans quatre cliniques différentes. Le but étant d'évaluer avec quel taux de précision ces enfants devaient obtenir le bon diagnostic, le résultat étant de 89% soit plus que ce qui était estimé au départ [11].

**Nouvelle méthode basée sur l'apprentissage Automatique** Il y a eu plusieurs études récentes, qui ont tentées de mettre en œuvre des approches de traitement de signaux plus évoluées pour améliorer la capacité de discerner l'TDAH [11]. Une étude menée par Mueller **and colleagues** ([41, [11]]) qui employait des méthodes d'apprentissage automatique sur un ICA (Independant Component Analysis) pour analyser les ERP et en extraire les critères d'un échantillon d'adultes TDAH et non TDAH. Une combinaison de 5 pointes d'amplitude, des mesures de la latence des ERPs, un monitoring et d'autres opérations sont faites afin de maximiser la discrimination TDAH au sein du groupe. Un taux élevé de (91%) dans la précision de la classification est obtenu pour l'échantillon original. ([41], [11]). D'autres études ont été faite dans le même domaine, se basant sur l'extraction des critères des EEG via un processus semi supervisé ([42], [11]).

**Limite des EEG** De nombreuses études limitent l'utilité des EEG dans la détermination de l'TDAH parmi un groupe de sujets. Cela est dû au fait que l'on doivent prendre en compte à la base un groupe de contrôle non-TDAH sans antécédent psychiatriques. Ainsi en mélangeant ce groupe avec un autre TDAH il devient en théorie plus facile de les distinguer, ce qui rend l'utilisation de l' EEG peu fiable [11]. De plus à cause du mixage spatial des signaux induits à la

fois par les électrodes et autres artefacts ne relevant pas de l'activité cérébrale (clignement des yeux, contraction musculaire...), une analyse par ordinateur est nécessaire pour séparer les signaux utiles des ondes étudiées, du bruit [11]. Toutefois, la naissance de nouvelles méthodes basées sur les outils du machines learning est une piste pour obtenir un diagnostic plus précis [11].

## 1.5 Deep learning et processus d'apprentissage

Dans le cadre des réseaux de neurones, le processus d'apprentissage peut s'assimiler à un processus dynamique et itératif permettant de modifier les paramètres d'un réseau en réaction avec les stimuli qu'il reçoit de son environnement [14]. Ce qui implique qu'un réseau se doit d'être stimulé par un environnement, qu'il subisse des modifications en réaction à cette stimulation, et que celles-ci provoquent dans le futur une nouvelle réponse vis à vis de l'environnement. Ainsi il peut améliorer ses performances. L'apprentissage consiste à changer les valeurs des poids reliant les neurones d'une couche à l'autre. Ce procédé est réalisé grâce à l'algorithme d'apprentissage.

### 1.5.1 Types d'apprentissage

On peut distinguer plusieurs tâches d'apprentissage telles que l'approximation, l'association, le classement, la prédiction et la commande. Le classement est la tâche qui sera développée dans la section suivante.

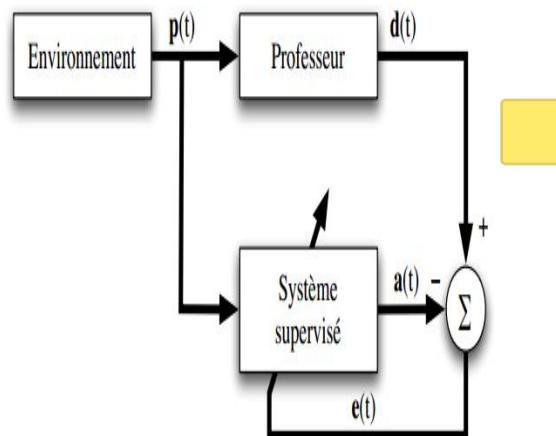


FIGURE 1.4 – Schéma bloc apprentissage supervisé [14]

### 1.5.1.1 Apprentissage supervisé

L'apprentissage dit supervisé est caractérisé par l'utilisation de données labellisées (exemples). En pratique, on a un ensemble de  $Q$  couples de vecteurs d'entrée et de sortie notés

$$(p_1, d_1), (p_2, d_2), \dots, (p_Q, d_Q), \quad \text{où } p_i \text{ désigne}$$

un stimulus (entrée) et  $d_i$  la cible pour ce stimulus, c'est-à-dire les sorties escomptées du réseau. Chaque couple  $(p_i, d_i)$  correspond donc à un cas d'espèce de ce que le réseau devrait produire (la cible) pour un stimulus donné. Pour cette raison, l'apprentissage supervisé est aussi qualifié d'apprentissage par des exemples (figure 1. 10). L'environnement est inconnu du réseau. Celui-ci produit un stimulus  $p$  qui est acheminé à la fois au bloc «professeur» et au réseau. Le professeur produit une sortie désirée  $d(t)$  pour ce stimulus. On suppose que cette réponse est optimale. Elle est ensuite comparée (par soustraction) avec la sortie du réseau pour produire un signal d'erreur  $e(t)$  qui est ré-injecté dans le réseau pour modifier son comportement de manière itérative qui, éventuellement, lui permet de simuler la réponse du professeur. Autrement dit, la connaissance de l'environnement par le professeur est graduellement transférée vers le réseau jusqu'à l'atteinte d'un certain critère d'arrêt. Ce transfert se fait soit via une régression linéaire soit via une régression logistique, cela en fonction des données du problème (nous parlerons des régressions dans la section suivante). Par la suite, on peut éliminer le professeur et laisser le réseau fonctionner de façon autonome [14].

### 1.5.1.2 Apprentissage non-supervisé

Il est caractérisé par l'absence complète de professeur, c'est-à-dire qu'on ne dispose ni d'un signal d'erreur, comme dans le cas supervisé, ni d'un indice de satisfaction, comme dans le cas par renforcement. Le réseau doit donc apprendre sans intervention externe. En assimilant les stimuli de l'environnement à une description de son état interne, la tâche du réseau est alors de modéliser cet état le mieux possible. Pour y arriver, il importe d'abord de définir une mesure de la qualité pour ce modèle, et de s'en servir par la suite pour optimiser les paramètres libres du réseau, c'est-à-dire ses poids synaptiques. A la fin de l'apprentissage, le réseau a développé une habilité à former des représentations internes des stimuli de l'environnement permettant d'encoder les caractéristiques de ceux-ci et, par conséquent, de créer automatiquement des classes de stimuli similaires. L'apprentissage non-supervisé s'appuie sur un processus compétitif [14].

## 1.5.2 Outils théoriques du deep learning

Le deep learning est une classe d'algorithme d'apprentissage automatique, basée sur les réseaux de neurones [19]. Il utilise des algorithmes d'apprentissage pour prédire une sortie  $Y$  en fonction d'une entrée  $X$  via une hypothèse  $H$  qui se fonde sur l'ensemble d'apprentissage de départ. Pour décrire le problème de l'apprentissage supervisé de manière formelle, étant donné

un ensemble d'apprentissage le but est d'apprendre une fonction  $h$  telle que  $h(x)$  soit un bon outil prédictif pour les valeurs correspondantes de  $y$ .

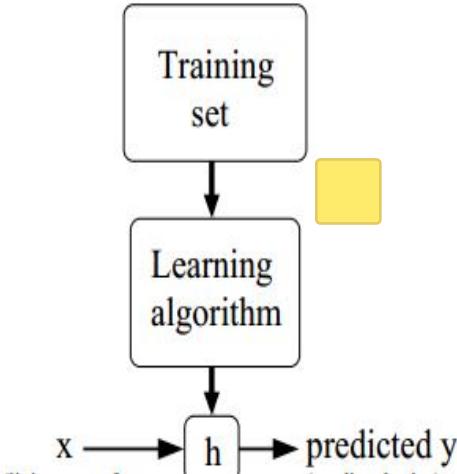


FIGURE 1.5 – Schéma bloc apprentissage [20]

Lorsque la variable cible que nous tentons de prédire est continue, on parle d'une régression linéaire et lorsqu'elle ne peut prendre qu'un petit nombre de valeurs discrètes on parle de classification [20].

### 1.5.2.1 Régression linéaire

On parle de régression linéaire lorsque la fonction de transfert d'une couche de neurone est une fonction linéaire. Le but de la régression linéaire est de prédire une valeur ciblée  $y$  partant d'un vecteur de valeurs d'entrées  $x$ . Il faut trouver une fonction  $y = h(x)$  (avec  $X$  et  $Y$  des sous ensembles de  $\mathbb{R}$  et leurs éléments sont continus) telle que pour chaque exemple entraîné

$$y_i \approx h(x_i)$$

Pour cela nous devons définir comment représenter  $h(x)$ . Posons

$$\sum_{i=0}^n \theta_i x_i = \theta^T x \quad \text{où } \theta \text{ représente les poids des connexions et } \theta_0 \text{ le biais et } x_0$$

N'est pas pris en compte car nous travaillons avec des vecteurs et

$$h_\theta(x) \text{ un ensemble de fonctions paramétrées selon } \theta$$

La tâche est de trouver  $\theta$  pour que

$$h(x_i) \text{ soit le plus proche possible de } y_i$$

En particulier on recherche un choix de thêta qui minimise la fonction objectif suivante :

$$J(\theta) = \frac{1}{2} \sum_{i=0}^m (h_\theta(x_i) - y_i)^2$$

Pour cela nous allons utiliser l'algorithme LMS (Least Mean Square) qui minimise l'erreur quadratique moyenne [20]. Il est basé sur le calcul du gradient descendant.

### 1.5.2.2 Régression logistique et classification

On parle de régression logistique lorsque la fonction de transfert d'une couche de neurones est une fonction sigmoïde. La régression logistique s'applique à des variables discrètes, et permet de prendre une décision binaire selon des valeurs de probabilités. On essaye de prédire la probabilité pour un exemple donné d'appartenir à la classe "1" contre la probabilité d'appartenir à la classe "0", plus précisément nous essayons d'apprendre une fonction de la forme :

$$P(y = 1|x) = h_\theta(x) = \frac{1}{1 + \exp(-\theta^T x)} \equiv \sigma(\theta^T x), \quad P(y = 0|x) = 1 - h_\theta(x). \text{ La fonction } \sigma(z) \equiv \frac{1}{1 + \exp(z)}$$

est une fonction sigmoïde, elle permet de ramener les valeurs de  $\theta^T x$

dans un intervalle [0,1] de sorte à interpréter

$h_\theta(x)$  comme une probabilité.

Le but est de trouver une valeur de theta telle que la probalité

$$P(y = 1|x) = h_\theta(x) \begin{cases} \text{"grande"} & \text{quand } x \text{ est dans la classe "1"} \\ \text{"petite"} & \text{quand } x \text{ est dans la classe "0" tel que } [P(y = 0|x) \text{ soit grand}] \end{cases}$$

Pour un ensemble d'exemples entraînés avec des éléments binaires, on cherche à trouver le choix de thêta qui va minimiser la fonction objectif suivante :

$$J(\theta) = - \sum_{i=0}^m \left( y_i \log(h_\theta(x_i)) + (1 - y_i) \log(1 - h_\theta(x_i)) \right)$$

Pour optimiser le choix de thêta on utilise également l'algorithme LMS, comme dans le cas de la régression linéaire [20].

**Algorithme LMS** La règle LMS consiste à tenter de minimiser un indice de performance F basé sur l'erreur quadratique moyenne. Possédant un ensemble d'apprentissage de Q associations stimulus/cible

Ceci nous permet de calculer l'erreur  $e(t)$  entre ce que le réseau produit en sortie pour ce stimulus et la cible  $d(t)$  qui lui est associée :

$$e(t) = d(t) - a(t). \tag{5.13}$$



Sachant que tous les neurones d'une même couche sont indépendants les uns des autres, et pour simplifier les équations, nous allons développer la règle LMS pour  $S = 1$ , c'est-à-dire le cas d'un seul neurone. Ensuite, nous pourrons facilement l'étendre au cas général de  $S$  neurones. Nous allons aussi regrouper tous les paramètres libres du neurone en un seul vecteur  $\mathbf{x}$  :

$$\mathbf{x} = \begin{bmatrix} \mathbf{w} \\ b \end{bmatrix}. \quad (5.14)$$

De même, nous allons regrouper en un vecteur  $\mathbf{y}$  le stimulus  $\mathbf{p}$  et l'entrée virtuelle  $-1$  associée au biais du neurone :

$$\mathbf{y} = \begin{bmatrix} \mathbf{p} \\ -1 \end{bmatrix}. \quad (5.15)$$

Ce qui nous permettra d'écrire la sortie  $a$  du neurone sous une forme simplifiée :

$$a = \mathbf{w}^T \mathbf{p} - b = \mathbf{x}^T \mathbf{y}. \quad (5.16)$$

Nous allons donc travailler avec le signal d'erreur scalaire  $e(t) = d(t) - a(t)$  et construire notre indice de performance  $F$  en fonction du vecteur  $\mathbf{x}$  des paramètres libres du neurone :

$$F(\mathbf{x}) = E [e^2(t)], \quad (5.17)$$

$$\hat{F}(\mathbf{x}) = e^2(t). \quad (5.18)$$

Alors, à chaque itération de l'algorithme, on peut calculer le vecteur gradient de cet estimé :

$$\hat{\nabla} F(\mathbf{x}) = \nabla e^2(t), \quad (5.19)$$

où les  $R$  premiers éléments de  $\nabla e^2(t)$  correspondent aux dérivés partielles par rapport aux  $R$  poids du neurone, et le dernier élément correspond à la dérivé partielle par rapport à son biais. Ainsi :

$$[\nabla e^2(t)]_j = \frac{\partial e^2(t)}{\partial w_{1,j}} = 2e(t) \frac{\partial e(t)}{\partial w_{1,j}}, \quad j = 1, \dots, R, \quad (5.20)$$

et :

$$[\nabla e^2(t)]_{R+1} = \frac{\partial e^2(t)}{\partial b} = 2e(t) \frac{\partial e(t)}{\partial b}. \quad (5.21)$$

Il s'agit maintenant de calculer les deux dérivés partielles de  $e(t)$  par rapport à  $w_{1,j}$  :

$$\frac{\partial e(t)}{\partial w_{1,j}} = \frac{\partial [d(t) - a(t)]}{\partial w_{1,j}} = \frac{\partial}{\partial w_{1,j}} [d(t) - (\mathbf{w}^T \mathbf{p}(t) - b_1)]$$

et  $b$  :

$$\frac{\partial e(t)}{\partial b} = 1. \quad (5.23)$$

Notez bien que les termes  $p_j(t)$  et  $-1$  sont les éléments de  $\mathbf{y}$ , de sorte qu'on peut écrire :

$$\hat{\nabla} F(\mathbf{x}) = \nabla e^2(t) = -2e(t)\mathbf{y}(t). \quad (5.24)$$

Ce résultat nous permet aussi d'apprécier la simplicité qu'engendre l'idée d'utiliser l'erreur instantanée plutôt que l'erreur moyenne. Pour calculer le gradient estimé de notre indice de performance, il suffit de multiplier l'erreur instantanée par le stimulus d'entrée !

L'équation 5.24 va nous permettre d'appliquer la méthode de la descente du gradient décrite par l'équation 4.7 (voir section 4.1, page 28) pour modifier les paramètres du neurone dans le sens d'une diminution de  $F$  :

$$\Delta \mathbf{x}(t) = \mathbf{x}(t+1) - \mathbf{x}(t) = -\eta \nabla F(\mathbf{x}) \Big|_{\mathbf{x}=\mathbf{x}(t)}. \quad (5.25)$$

En substituant  $\nabla F(\mathbf{x})$  par  $\hat{\nabla} F(\mathbf{x})$ , on obtient :

$$\Delta \mathbf{x}(t) = 2\eta e(t) \mathbf{y}(t), \quad (5.26)$$

ce qui équivaut à :

$$\Delta \mathbf{w}(t) = 2\eta e(t) \mathbf{p}(t), \quad (5.27)$$

$$\Delta b(t) = -2\eta e(t). \quad (5.28)$$

Les équations 5.27 et 5.28 définissent la règle LMS de base. On la nomme également règle de Widrow-Hoff, du nom de ses auteurs. Dans le cas d'une couche de  $S$  neurones, nous pourrons mettre à jour chaque rangée  $i$  de la matrice de poids ainsi que chaque élément  $i$  du vecteur de biais à l'aide des équations suivantes :

$$\Delta_i \mathbf{w}(t) = 2\eta e_i(t) \mathbf{p}(t), \quad (5.29)$$

$$\Delta b_i(t) = -2\eta e_i(t). \quad (5.30)$$

Ce qui nous permet de réécrire le tout sous la forme matricielle :

$$\Delta \mathbf{W}(t) = 2\eta \mathbf{e}(t) \mathbf{p}^T(t), \quad (5.31)$$

$$\Delta \mathbf{b}(t) = -2\eta \mathbf{e}(t). \quad (5.32)$$

La méthode de la descente du gradient est garantie de converger vers un minimum global, a condition de restreindre la valeur du taux d'apprentissage [14]

### 1.5.2.3 Les réseaux de neurones

Formés de structures cellulaires artificielles, ils constituent une approche permettant d'aborder sous des angles différents les problèmes de perception, de mémoire, d'apprentissage et de raisonnement. Ils s'avèrent aussi être des alternatives très prometteuses pour contourner certaines des limitations des ordinateurs classiques. Grâce à leurs mécanismes inspirés des cellules nerveuses (neurones), ils infèrent des propriétés émergentes permettant de solutionner des problèmes autrefois qualifiés de complexes [19]. Ils sont modélisés par un maillage de plusieurs neurones artificiels, organisés en couches.

**Quelques notations** Les notations qui suivent seront utilisées par la suite pour la modélisation mathématique d'un réseau de neurones [14].

#### Poids d'une couche de neurones

- $W_k(t)$  désigne la matrice des poids pour la couche  $k$  d'un réseau au temps  $t$ .
- $w_{kj}(t)$  désigne le vecteur correspondant à la colonne  $j$  de  $W_k(t)$ . –  $i_{wk}(t)$  désigne le vecteur-rangée correspondant à la ligne  $i$  de  $W_k(t)$ .
- $w_{ki,j}(t)$  désigne l'élément  $(i,j)$  de  $W_k(t)$  ( $i$  désigne toujours une ligne et  $j$  une colonne).

#### Biais d'une couche de neurones

- $b_k(t)$  désigne le vecteur des biais pour la couche  $k$  d'un réseau au temps  $t$ .
- $b_{ki}(t)$  désigne l'élément  $i$  de  $b_k(t)$ .

**Stimulus d'un réseau**

- $p(t)$  désigne un vecteur stimulus présenté à l'entrée d'un réseau au temps  $t$ .
- $p_i(t)$  désigne l' élément  $i$  de  $p(t)$ .

**Niveaux d'activation d'une couche de neurones**

- $n_k(t)$  designe le vecteur des niveaux d'activation pour la couche  $k$  d'un réseau au temps  $t$ .
- $n_{ki}(t)$  designe l' élément  $i$  de  $n_k(t)$ .

**Sorties d'une couche de neurones**

- $a_k(t)$  désigne un vecteur des sorties pour la couche  $k$  d'un réseau au temps  $t$ .
- $a_{ki}(t)$  désigne l' élément  $i$  de  $a_k(t)$ .

**Cibles d'un réseau**

- $d(t)$  désigne un vecteur cible pour les sorties désirées d'un réseau au temps  $t$ .
- $d_i(t)$  désigne l'élément  $i$  de  $d(t)$ .

**Base d'apprentissage**

- $(p_1, d_1), (p_2, d_2), \dots, (p_Q, d_Q)$  désigne un ensemble de  $Q$  associations stimulus/cible pour l'apprentissage supervisé.

**Signaux d'erreur**

- $e(t) = [d(t) - a(t)]$  désigne un vecteur mesurant l'erreur entre les sorties désirées (cible) et les sorties calculées d'un réseau au temps  $t$ .
- $e_i(t)$  désigne l' élément  $i$  de  $e(t)$ .

**Dimensions**

- $M$  désigne le nombre de couches d'un réseau. –  $S_k$  désigne le nombre de neurones sur la couche  $k$  d'un réseau.
- $Q$  désigne le nombre d'associations pour l'apprentissage.
- $R$  désigne la dimension des stimulus d'entrée.

**Fonctions de transfert d'une couche de neurones**

- $f_k(n_k) = a_k$  désigne le vecteur des sorties de la couche  $k$ , telles que calculées par la fonction de transfert  $f$  appliquée sur chacun des  $n_k$  i ,  $i = 1, \dots, S_k$ .

**Modèle d'un neurone artificiel :** Un neurone est essentiellement constitué d'un intégrateur qui effectue la somme pondérée de ses entrées. Le résultat  $n$  de cette somme est ensuite transformé

## CHAPITRE 1. INTRODUCTION

---

par une fonction de transfert  $f$  qui produit la sortie  $a$  du neurone. En suivant les notations présentées précédemment, les  $R$  entrées du neurones correspondent au vecteur  $p = [p_1 p_2 \dots p_R]^T$ , alors que  $w = [w_1, w_1, 2 \dots w_1, R]^T$  représente le vecteur des poids du neurone. La sortie  $n$  de l'intégrateur est donnée par l'équation suivante :  $a = f([dWp - b])$ .

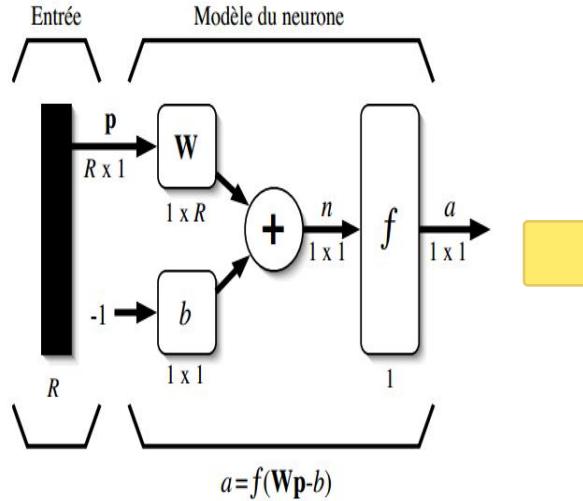


FIGURE 1.6 – Neurone artificiel , [14]

Une couche  $k$  de  $S$  neurones représente donc un ensemble de neurones agencés de la manière illustrée à la figure 1.9. Un réseau de neurones peut donc se définir comme une succession de couches reliant un vecteur d'entrée  $R$  à un vecteur de sortie  $a$ .

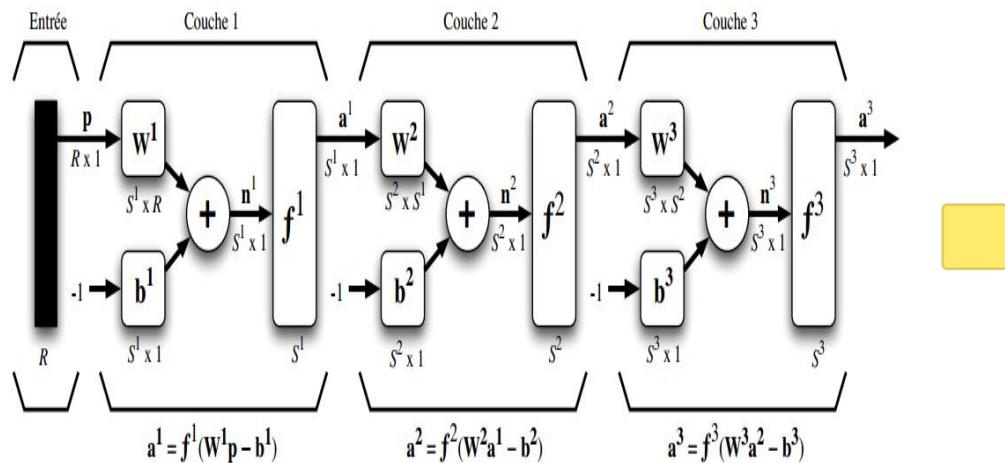


FIGURE 1.7 – Réseau de 3 Couches de  $S$  neurones , [14]

Parmi les propriétés escomptées pour un réseau de neurones, la plus importante est la

capacité d'apprendre de son environnement, d'améliorer sa performance à travers un processus d'apprentissage.

#### 1.5.2.4 Réseaux de neurones multicouches

Les réseaux multicouches permettent de résoudre des problèmes de classification linéairement inséparables. On peut même démontrer qu'avec un réseau de trois couches (deux couches cachées + une couche de sortie), comme celui de la figure précédente on peut construire des frontières de décision de complexité quelconque à condition d'employer une fonction de transfert non linéaire et de disposer d'assez de neurones sur les couches cachées [14]. Un réseau multicouche est un assemblage de couches concaténées les unes aux autre, de la gauche vers la droite en prenant les sorties d'une couche et en les injectant comme les entrées de la couche suivante. L'algorithme de rétropropagation vu à la section qui suit est l'algorithme utilisé pour entraîner un réseau de neurones multicouches.

Pour faire de la classification on utilisera des réseaux soit à deux soit à trois couches de neurones sigmoïdes. On peut montrer qu'une seule couche cachée suffit à engendrer des frontières de décision convexes, ouvertes ou fermées, de complexité arbitraire, alors que deux couches cachées permettent de créer des frontières de décision concaves ou convexes, ouvertes ou fermées, de complexité arbitraire. La première couche cachée d'un tel réseau sert à découper l'espace d'entrée à l'aide de frontières de décision linéaires, la deuxième couche sert à assembler des frontières de décision non-linéaires convexes en sélectionnant ou en retranchant des régions engendrées par la couche précédente et, de même, la couche de sortie permet d'assembler des frontières de décision concaves en sélectionnant ou en retranchant des régions convexes engendrées par la couche précédente [14].

#### 1.5.3 Algorithme de rétro-propagation

Pour qu'un apprentissage entre plusieurs couches d'un réseau se fasse de manière automatique, il est nécessaire d'utiliser un algorithme de rétro-propagation des erreurs. Soit le couple un couple  $(P_n, D_n)$  représentant la  $n$ ième donnée d'entraînement du réseau c'est-à-dire contenant un vecteur d'entrées et un vecteur de sorties désirées, où  $P$  et  $D$  sont les vecteurs. L'algorithme de rétro-propagation procède à l'adaptation des poids neurone par neurone en partant de la couche de sortie. L'algorithme de rétro-propagation standard se résume à la série d'étapes suivantes :

1. Initialiser tous les poids à de petites valeurs aléatoires dans l'intervalle (-0.5 , 0.5);
2. Normaliser les données d'entraînements ;
3. Permuter aléatoirement les données d'entraînement ;
4. Pour chaque donnée d'entraînement  $n$  :
  - a) Calculer les sorties observées en propageant les entrées vers l'avant ;

b) Ajuster les poids en rétro-propageant l'erreur observée.

$$w_{ji}(n) = w_{ji}(n - 1) + \Delta w_{ji}(n) = w_{ji}(n - 1) + \eta \delta_j(n) y_i(n)$$

où le “gradient local” est défini par :

$$\delta_j(n) = \begin{cases} e_j(n) y_j(n)[1 - y_j(n)] & \text{Si } j \in \text{couche de sortie} \\ y_j(n)[1 - y_j(n)] \sum_k \delta_k(n) w_{kj}(n) & \text{Si } j \in \text{couche cachée} \end{cases}$$



avec n inclus dans [0,1] représentant le taux d'apprentissage et  $y_i(n)$  représentant soit la sortie du neurone i sur la couche précédente, si celui-ci existe, soit l'entrée i autrement.

5. Répéter les étapes 3 et 4 jusqu'à un nombre maximum d'itération ou jusqu'à ce que la racine de l'erreur quadratique moyenne (EQM) soit inférieure à un certain seuil [13].

#### 1.5.4 Types d'architectures

Il existe une très grande variété d'architectures de deep learning. Il n'est pas toujours possible de comparer les performances des architectures entre elles car elles ne sont pas toutes évaluées sur le même ensemble de données. On distingue les grandes classes d'architectures telles que le deep neural network, le convolutional deep neural network, le deep belief network et le recurrent neural network. Chacune de ces architectures permet d'effectuer des tâches spécifiques, et suivant le type que l'on met en œuvre, le temps d'exécution de l'apprentissage peut être plus ou moins long de l'ordre de quelques jours à quelques heures. Il faut noter que ce temps d'exécution a diminué depuis le début des années 2000 grâce à l'évolution matérielle des ordinateurs, plus précisément des processeurs graphiques (ces derniers pouvant traiter des matrices/vecteurs de données plus larges et ce de manière plus efficace et rapide).



**Deep neural networks** Le Deep Neural Network (DNN) est un réseau neuronal artificiel comportant plusieurs couches cachées entre les couches d'entrée et de sortie. Il peut modéliser des relations non linéaires complexes. Cette architecture génère des modèles composés où les objets sont représentés comme des superpositions d'images primitives. Il est typiquement modélisé comme un réseau "feedforward" où les informations ne circulent que dans une seule direction ; des nœuds d'entrée au travers des nœuds cachés s'ils existent jusqu'aux nœuds de sortie, il n'y a donc aucun cycle dans le réseau. C'est le premier et le plus simple type de réseaux artificiel de neurones. Il est utilisé dans la détection d'objets [2]. Un réseau DNN peut être entraîné de manière discriminante en utilisant l'algorithme standard de rétro-propagation.

**Convolutionnal neural networks** Le Convolutional Neural Network est une méthode utilisée pour le traitement des données visuelles ainsi que celle à deux dimensions. Il est composé d'une ou de plusieurs couches de convolutions où toutes les couches sont connectées à l'entrée (couche d'entrée). Cette architecture utilise une pondération uniforme des couches et une couche de mise en commun des données. En comparaison avec d'autres architectures, CNN est celle qui a

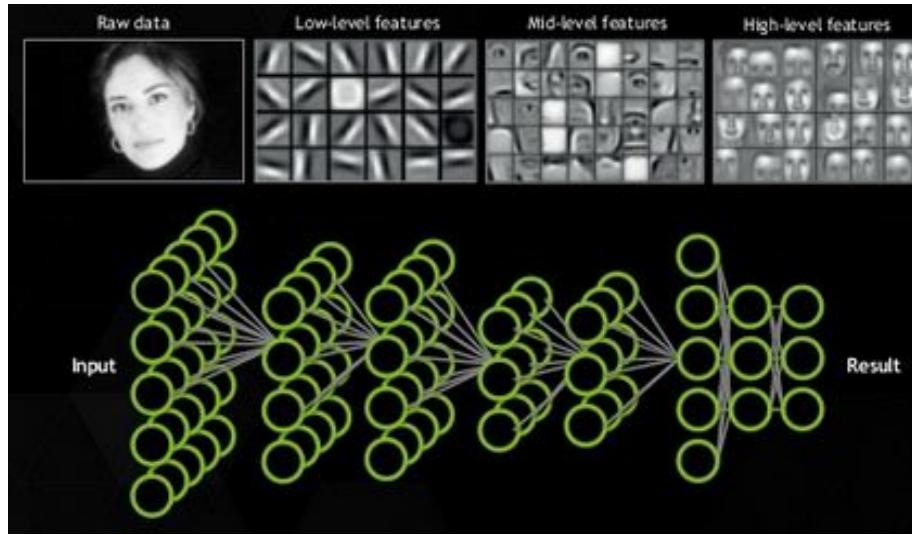


FIGURE 1.8 – Deep Neural Network , [27]

fournie de meilleurs résultats dans le domaine de la reconnaissance d'image et du langage [3]. Ils peuvent également être entraînés en utilisant l'algorithme standard de rétro-propagation.

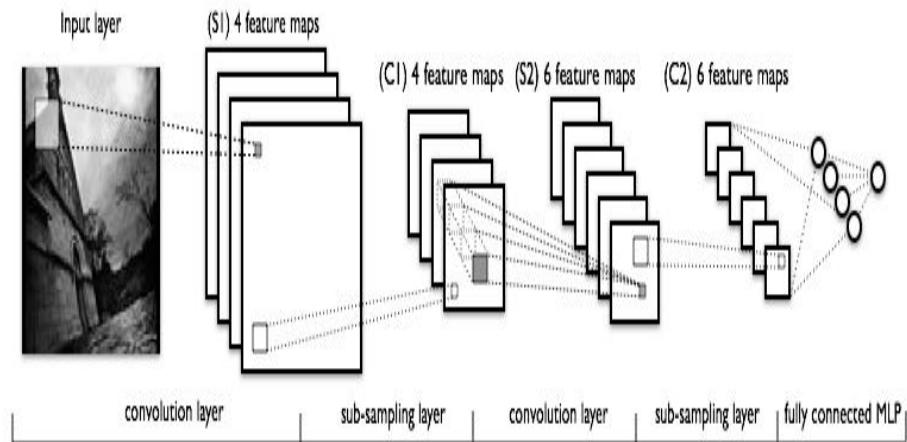


FIGURE 1.9 – Convolutional Neural Network [3]

**Recursive Neural Networks** Le Réseau récursif neuronal (RNN) est créé en appliquant le même ensemble de poids sur une structure formant un graphe dirigé acyclique. Il est assimilable au Recurrent Neural Network mais à la seule différence que le recurrent Neural Network est profond dans le temps, le RNN lui est profond dans sa structure à cause de l'application répétée des connexions récursives. Il est appliqué dans les processus de traitement du langage naturel [18].

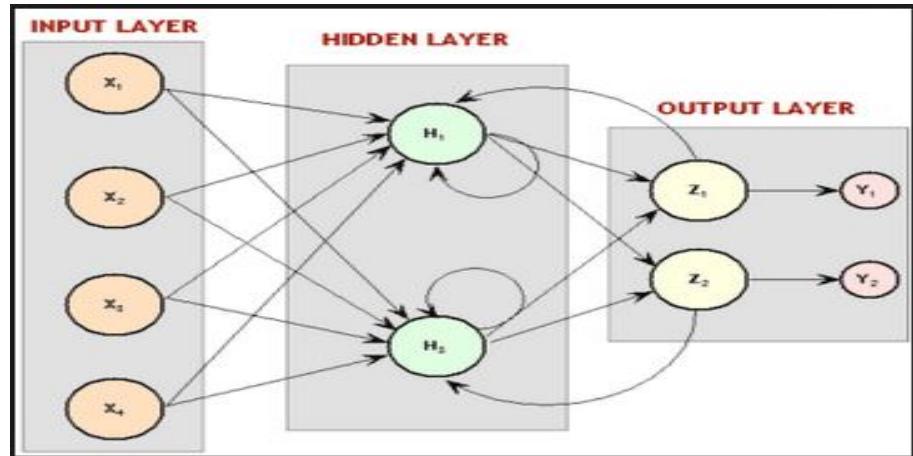


FIGURE 1.10 – Recursive Neural Network [7]

**Deep Belief Networks** Le réseau de deep belief (DBN) est composé d'une multitude de couches de variables stochastiques et latentes. Ces variables sont habituellement binaires et sont souvent appelés unités cachées ou détecteurs de fonctionnalités. L'idée majeure qui se cache derrière cette architecture est qu'elle est dotée d'une procédure d'apprentissage couche par couche allant du haut vers le bas. Supposons l'apprentissage d'un ensemble d'exemples entraînés de manière non supervisée, le DBN peut via les probabilités reconstruire ses entrées ; Les couches vont donc se comporter comme des détecteurs de fonctionnalités sur les entrées. Le DBN peut après cette étape d'apprentissage, être entraîné cette fois de manière non supervisée pour procéder à une classification de données. Il est considéré comme une composition de sous réseaux formant une Machine de Boltzmann réduite, où chaque sous réseaux de couches cachées sert de couche visible pour le suivant [3].

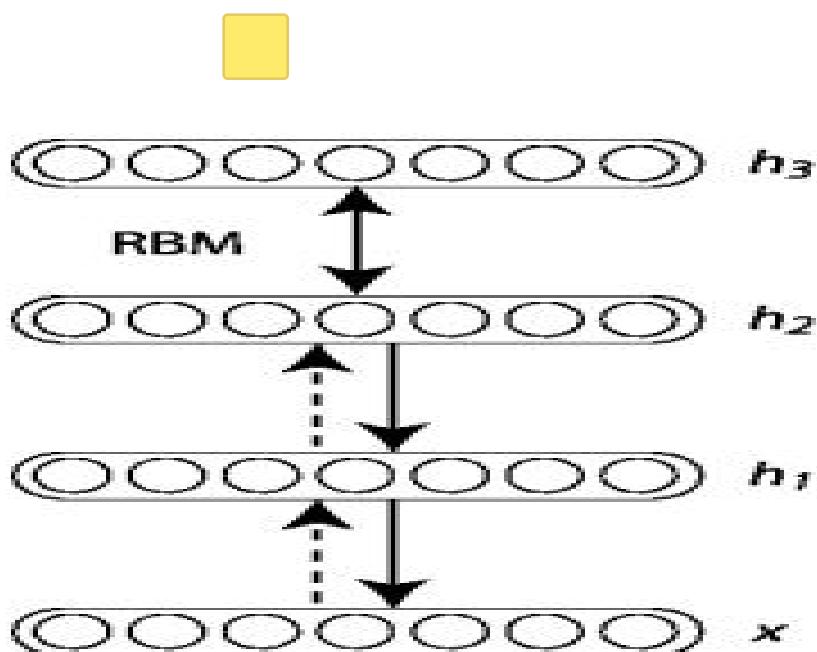


FIGURE 1.11 – Deep Belief Network [3]



## BUT DU TRAVAIL

### **2.1 projet de mise en œuvre**

Le but est de pouvoir automatiser le diagnostic de la pathologie TDAH en utilisant le deep learning basé sur des acquisitions EEG. C'est-à-dire que l'on doit après l'analyse d'un EEG d'un patient, dire si oui ou non il est TDAH. En d'autres termes faire un classement des EEG suivant deux catégories. La tâche d'apprentissage du réseau sera la classification.

#### **2.1.1 Tâche d'apprentissage**

La base d'apprentissage est représentée ici par un ensemble de relevés EEG (stimulus) et les patterns d' EEG TDAH (cible 1), et d' EEG non TDAH (cible 2).

Pour cette tâche, il existe un nombre fixe de catégories d'entrée que le réseau doit apprendre à reconnaître. Dans un premier temps, le réseau doit entreprendre une phase d'apprentissage supervisé durant laquelle les stimuli sont présentés en entrée et les catégories sont utilisées pour former les sorties désirées, généralement en utilisant une sortie par catégorie. Ainsi, la sortie 1 est associée à la catégorie 1, la sortie 2 à la catégorie 2, etc. Notre problème comportant 2 catégories, on peut par exemple fixer les sorties désirées

$$d = [d1, d2]^T$$

à l'aide de l'expression suivante :

$$d(i) = \begin{cases} 1 & \text{si le stimulus appartient à la catégorie 1 TDAH} \\ 0 & \text{si le stimulus appartient à la catégorie 2 non TDAH} \end{cases} \quad \forall i, \in [1, 2].$$

Par la suite, dans une phase de classification, il suffira de présenter au réseau n'importe quel stimulus inconnu pour pouvoir procéder à la classification de celui-ci dans l'une ou l'autre des catégories. Une règle simple de classification consiste, par exemple, à choisir la catégorie associée avec la sortie maximale.

### 2.1.2 Classificateur et couches

Nous devons trouver un classificateur qui soit adapté à cet effet. Nous devons déterminer le nombre de couches (couches cachées + couche de sortie), ainsi que le nombre de neurones par couche. Le PMC (Perceptron Multicouche) est un classificateur à plusieurs couches permettant de procéder à un apprentissage automatique du réseau, car les fonctions de transfert des couches cachées sont non linéaires pour que le réseau puisse résoudre des problèmes complexes où la frontière de décision entre les classes n'est pas linéaire (Fonction de transfert de type sigmoïde [14]).

$$\text{sigmoïde} \quad \parallel \quad a = \frac{1}{1+\exp^{-n}} \quad \boxed{\text{Sigmoid Function Graph}} \quad \boxed{\text{Yellow Square}}$$

Il faut aussi déterminer via des calculs appropriés le poids des connexions entre chaque entité du vecteur d'entrée et les neurones de la première couche, et de manière itérative les poids des connexions entre les sorties d'une couche inférieure et les neurones d'une couche supérieure.

Nous avons vu des types d'architecture à la section 1. 2. 1. 4. Notre choix se portera sur l'implémentation du DBN. Ce choix est justifié par la nature du problème qui est traité, et aussi au vu des recherches et cas d'utilisation de ce type d'architecture référencée dans l'état de l'art. Ce dernier sera associé à un auto-encodeur pour extraire les critères utiles des ERPs et un SVM (Support Vector Machine) qui recevra en entrée les données venant de l'auto-encodeur.

### 2.1.3 Approche de résolution

Une fois la structure complète du réseau faite (voir section précédente), un algorithme de résolution serait le suivant :

1. Récupérer un ensemble d'échantillons d'EEG.
2. Les faire analyser par un auto encodeur et extraire les critères (ratio theta/ bêta, moyenne ondes theta et bêta).
3. Passer en entrée tous les critères extraits au réseau multicouches.
4. Appliquer une reconnaissance de pattern sur les entrée.
5. Regarder si la courbe analysée approche celle de l'une des catégories.
6. Procéder à la classification.



## MATÉRIEL ET MÉTHODES

En se basant sur une expérimentation faite au laboratoire de Human Waves, l'on constate que l'activité électrique liée à la tâche cognitive que le sujet est en train d'accomplir peut être très faible et se fondre dans l'activité spontanée (liée au échanges naturels de signaux entre les neurones), de sorte qu'il n'est possible de l'extraire qu'en faisant plusieurs essais suivant la même tâche. Le matériel utilisé pour faire les enregistrements est tout d'abord le bonnet portant les 128 électrodes (les données EEG manipulées seront des EEG multicanal avec 128 canaux) qui vont se charger de capter la différence de potentiel dans les régions du cerveau aux alentours de l'électrode. Un dispositif de conversion analogique numérique va permettre ensuite de transférer les signaux électriques reçus vers un ordinateur, sur lequel tourne un logiciel adapté. La figure suivante est une illustration du port d'un bonnet lors d'une expérimentation.



FIGURE 3.1 – Bonnet porté par une patiente, Human Waves

### 3.1 Méthodologie

Notons que toutes les données et informations présentes dans cette section proviennent du responsable du projet TDA/H, de Human Waves. La première étape est le recrutement des enfants dans une tranche d'âge de 6 à 18 ans, avec leur consentement et celui de leur parents (ou juste celui du parent accompagnateur).

**Patients et contrôles** Une quarantaine d'enfants dont 13 sont non TDAH et forment le groupe de contrôles, 20 sont TDAH et 7 dont on souhaite statuer sur leur état TDAH ou non. c'est le groupe des patients.

**Données comportementales du patient** Les données relatant le comportement des patients, dans divers environnements comme au domicile familial, à l'école, dans les salles de jeux ; ou dans certaines situations comme des disputes, des moments de concentration, de décision sont encodés sur des fiches. Ces fiches permettront de voir si le patient à une tendance à avoir la pathologie. Celles ci seront ensuite ajoutées aux données expérimentales obtenues pour rendre un avis plus précis sur le statut du sujet. De plus des interviews cliniques avec le médecin de famille peuvent les accompagner.

**Le protocole CUE-GO/NO GO** C'est un procédé de contrôle suivant lequel un stimulus modélisé par un carré : il s'agit de la phase « CUE », est présenté dans un flux continu au participant. Après l'apparition du Q le sujet doit émettre une décision binaire. Concrètement ce qui se passe lors d'une expérimentation chez Human Waves est que sur un intervalle de temps de 4 minutes et 20 secondes l'on soumet au participant un flux contenant des formes et des symboles.

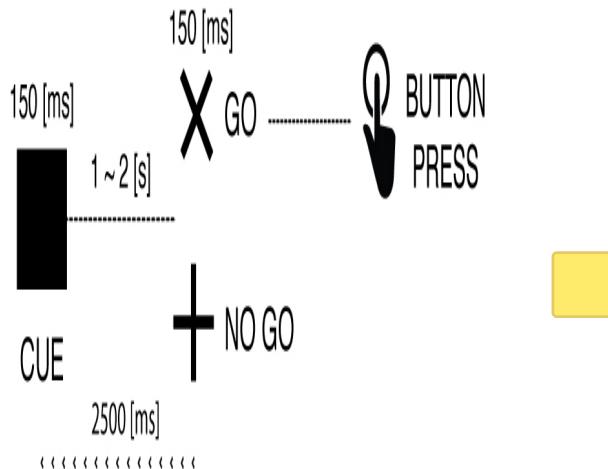


FIGURE 3.2 – Protocole Q GO/NO GO, source Human Wavesaves

Le processus se fait via un ordinateur. Au départ un carré apparaît, après ce dernier si le symbole est une croix « + », il doit faire une action (go) et appuyer la touche « Espace » sur le clavier et si le symbole est une croix « x » il ne doit rien faire (no go). L'exactitude et le temps de réaction sont mesurés pour chaque événement. Sur des statistiques faites, Il apparaît que les événements liés au «go» surviennent avec une fréquence plus élevée (environ 80%) que ceux liés au «no go» (environ 20%) [10]. L'activité cérébrale étant enregistrée en parallèle lors de la réalisation de ces tâches cognitives , les membres de l'équipe de développement de NeuroAtt (Projet lié à l'aide au diagnostic clinique de la TDA/H) sont en mesure d'évaluer la probabilité du patient d'être atteint par la maladie au moyen de différents critères physiologiques (analyse des rythmes cérébraux et le temps de réaction des patients).

**Analyse des EEG** Les événements sont enregistrés sur chacune des électrodes du scalp pour chaque patient. Puis un découpage en époque est fait en fonction du **q**, du go et du no go. on obtient donc au final trois ensembles séparés. Pour chacun de ces ensembles une analyse statistique est réalisée sur la moyenne des ERP. De plus le ratio thêta / bêta est comparés entre les contrôles et les patients lors de la mesure de l' ERP lié à la tâche GO/NOGO. (Figure 1.6)

## CHAPITRE 3. MATÉRIEL ET MÉTHODES

---

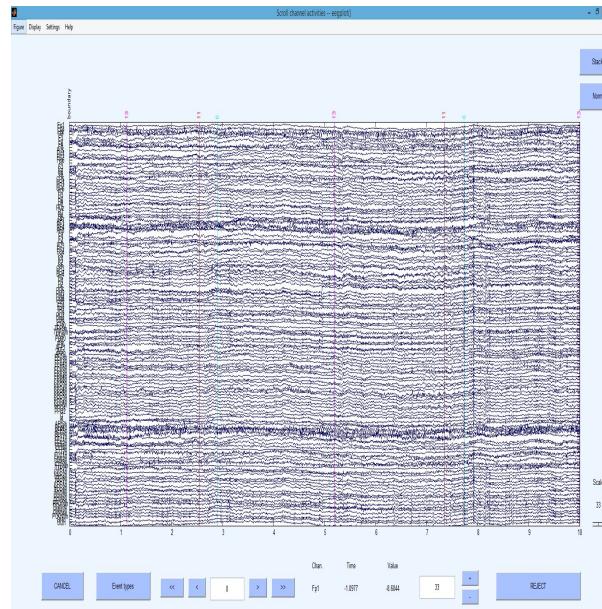


FIGURE 3.3 – Epoque de Q, source Human Wavesaves

**Analyse statistique** Il s'agit du dernier traitement que subissent les données extraites des EEG. Toutes les informations des sujets sont regroupées par époque dans une data-set, puis une analyse statistique par permutations est faite. On choisit sur chaque sujet une électrode sur laquelle on voudrait faire la moyenne de ERP, puis on compare la courbe obtenue avec celle des TDA/H pour en tirer une conclusion. Un exemple de courbe est donnée dans la figure 1.7.

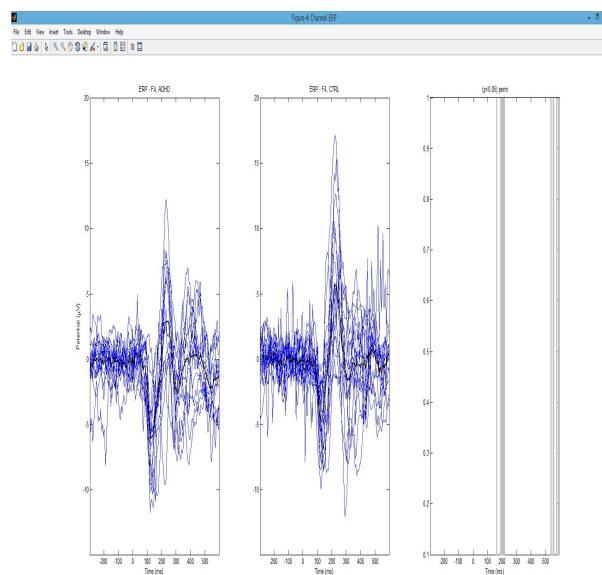


FIGURE 3.4 – Moyenne ERP contrôle et TDAH, source Human Wavesaves

## 3.2 Projet de réalisation

Dans cette section, des exemples de logiciels existant permettant de faire du deep learning. Des cas d'exemples de langage de programmation adaptés.

### 3.2.1 Logiciels et outils existants

On peut citer des outils et librairies tels que CNTK (Computer Network ToolKit) de microsoft, Theano, TensorFlow de google, Torch7 et Caffe. Pour certain d'entre eux le langage de programmation est le python.

**TensorFlow** C'est un logiciel de librairie open source pour des calculs numériques utilisant des données sous forme de diagramme de flux. La flexibilité de son architecture permet de déployer des calculs sur plus d'un CPU ou GPU sur un desktop, un serveur, un mobile [tensorflow.org].

**Torch**

## 3.3 Conclusion



## RÉSULTATS



## DISCUSSION ET CONCLUSION GÉNÉRALE



C H A P I T R E



RÉFÉRENCES



## BIBLIOGRAPHIE

- [1] ARSEP, *Pathologie*.
- [2] J.-P. BANQUET, *Spectral analysis of the eeg in meditation*, *Electroencephalography and clinical neurophysiology*, 35 (1973), pp. 143–151.
- [3] DEEPMLEARNING, *tutorial* .
- [4] J. FLUSS AND C. KORFF, *Place de l'eeg dans le diagnostic des troubles neurodéveloppementaux*, *Neuropsychiatrie de l'Enfance et de l'Adolescence*, 59 (2011), pp. 369 – 375.
- [5] P. GALLOIS, G. FORZY, J. LEDUC, F. ANDRES, L. PEYRODIE, E. LEFEBVRE, AND P. HAU-TECOEUR, *Comparaison de l'analyse spectrale et de l'analyse non-linéaire de l'eeg dans les troubles cognitifs*, *Neurophysiologie Clinique/Clinical Neurophysiology*, 32 (2002), pp. 297 – 302.
- [6] L. GARNERO, *Imagerie cérébrale fonctionnelle : Techniques et applications*, Laboratoire de Neurosciences Cognitives& Imagerie Cérébrale, CNRS UPR640. Centre de Magnétoencéphalographie.
- [7] L. GARNERO, *Les bases physiques et physiologiques de la magnétoencéphalographie et de l'electroencéphalographie*, tech. rep., Citeseer, 2001.
- [8] HUMANWAVES, *TDAH* .
- [9] S. JIRAYUCHAROENSAK, S. PAN-NGUM, AND P. ISRASENA, *Eeg-based emotion recognition using deep learning network with principal component based covariate shift adaptation*, *The Scientific World Journal*, 2014 (2014).
- [10] S. J. JOHNSTONE, R. J. BARRY, AND A. R. CLARKE, *Ten years on : a follow-up review of erp research in attention-deficit / hyperactivity disorder*, *Clinical Neurophysiology*, 124 (2013), pp. 644–657.
- [11] S. K. LOO AND S. MAKEIG, *Clinical utility of eeg in attention-deficit / hyperactivity disorder : a research update*, *Neurotherapeutics*, 9 (2012), pp. 569–587.

## BIBLIOGRAPHIE

---

- [12] G. OGRIM, J. KROPOTOV, AND K. HESTAD, *The quantitative eeg theta / beta ratio in attention deficit/hyperactivity disorder and normal controls : sensitivity, specificity, and behavioral correlates*, Psychiatry research, 198 (2012), pp. 482–488.
- [13] M. PARIZEAU, *Le perceptron multicouche et son algorithme de rétropropagation des erreurs*, département de génie électrique et de génie informatique, Université de laval, (2004).
- [14] ———, *Réseaux de neurones*, GIF-21140 et GIF-64326, 124 (2004).
- [15] A. RAKOTOMAMONJY, V. GUIQUE, G. MALLET, AND V. ALVARADO, *Classification d'eeg pour les interfaces cerveau-machine*, in 20° Colloque sur le traitement du signal et des images, FRA, 2005, GRETSI, Groupe d'Etudes du Traitement du Signal et des Images, 2005.
- [16] M. RIO, *Bayesian models for synchronizations detection in electrocortical signals*, theses, Université de Lorraine, July 2013.
- [17] A. SALVAIL-BÉRARD, *Réseaux de neurones*, 13 (2012).
- [18] C. SZEGEDY, A. TOSHEV, AND D. ERHAN, *Deep neural networks for object detection*, in Advances in Neural Information Processing Systems, 2013, pp. 2553–2561.
- [19] TECHNOLOGIES EBUSINESS, *deep learning*.
- [20] S. UNIVERSITY, *tutorial*.
- [21] D. WULSIN, J. GUPTA, R. MANI, J. BLANCO, AND B. LITT, *Modeling electroencephalography waveforms with semi-supervised deep belief nets : fast classification and anomaly measurement*, Journal of neural engineering, 8 (2011), p. 036015.