République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université 8Mai 1945 – Guelma
Faculté des sciences et de la Technologie
Département d'Electronique et Télécommunications

Mémoire de fin d'étude
pour l'obtention du diplôme de Master Académique

Domaine : **Sciences et Technologie**
Filière : **Electronique**
Spécialité : **Systèmes Electroniques**

Analyse des données par les réseaux de neurones de type cartes auto-organisatrices de Kohonen

---

Présenté par :

BOUROUGA Khayreddine
BAZINE Khaled

Sous la direction de :

Dr. Mohamed Nemissi

Juin 2016

Remerciements

Mes remerciements vont en premier lieu à ALLAH Tout Puissant qui a illuminé mon chemin de la lueur du savoir et de la science et pour la volonté, la santé et la patience qu'il ma prodiguées durant toutes ces années d'études.

Je tiens aussi à exprimer ma reconnaissance et ma profonde gratitude à Monsieur Mohamed Nemissi, Dr à l'université 8 mai 1945 de Guelma, pour avoir Assuré l'encadrement de ce travail. Son aide, sa grande disponibilité ont joué un Rôle essentiel dans l'aboutissement de ce travail.

Enfin, mes remerciements vont aussi à mes parents pour leur patience, leurs Encouragements continus et leur soutien inconditionnel.

Dédicace

Je tiens à remercier en premier lieu Allah qui m'a donné, vie et santé pour le parachèvement de ce modeste ouvrage.

C'est avec profonde gratitude et sincères mots, que je dédier ce fameux travail de fin d'étude

Aux deux êtres les plus chers au monde, qui ont souffert nuit et jour pour nous couvrir de leur chaleur d'amour, mes parents.

L'être qui me guide dans ma vie et que j'imite son honnêteté, son sérieux et sa responsabilité de ces engagements, mon cher père.

A ma source de bonheur, la perle de mes yeux, ma mère.

Que dieu vous garde en bonne santé.

A mes chers frères et sœurs.

A toutes ma famille de Bazine et Bourouga.

A tous mes amis

A mon Encadreur Dr, Mohamed Nemissi

A mes camarades de filière systèmes électroniques

A toutes la promo 2015- 2016

Sommaire

Chapitre II : Clustering par SOM

Chapitre III : Application

Liste des figures

Liste des tableaux

Introduction générale

De nos jours, les données stockées sous forme numérique ne cessent de croitre de plus en plus partout dans le monde et dans tous les domaines. Les chercheurs, les scientifiques, les industriels...etc. Mettent de plus en plus leurs informations à la disposition de tout le monde. De nombreuses mesures effectuées un peu partout permettent la création de bases de données numériques énormes. Il est donc important de développer des techniques permettant d'utiliser aux mieux tous ces stocks d'informations, tel que la classification automatique afin d'en extraire les connaissances utiles.

Les cartes auto-organisatrices de Kohonen (SOM, Self-Organizing Maps) sont largement utilisées comme méthodes d'analyse et de visualisation de données complexes incluant de très grand rang d'applications dans différents domaines. Ces cartes font partie des réseaux de neurones dont la caractéristique principale est leur capacité de représenter les relations non linéaire d'un ensemble de données multidimensionnelles dans une grille de neurones à une, deux ou trois dimensions facilement visualisables.

L'objectif de ce mémoire consiste à étudier les capacités de la classification non supervisée basée sur les cartes auto-organisatrices de Kohonen et leurs capacités d'analyse des données biomédicale. Nous effectuons des tests sur la base de données cancers humain qui comporte 6 classes dont trois classes représentent des tissus normaux et trois classes représentent des tissus pathologiques. L'idée de base derrière la réalisation de cette base de données repose sur le fait que la conduction électrique dans un tissu peut être modifiée par les changements qui se produisent comme la présence d'une lésion ou d'une tumeur. Les caractéristiques de cette base de données se basent donc sur des mesures d'impédance qui ont été faites à différentes fréquences. Le présent mémoire comporte trois chapitres :

Le premier chapitre consiste en une introduction aux réseaux de neurones. Nous décrivons leurs principes de modélisation, leur apprentissage et leurs architectures. Nous présentons également à la fin de ce chapitre quelques modèles de base de réseaux de neurones.

Le deuxième chapitre concerne le clustering par les cartes auto-organisatrices de Kohonen. Nous donnons au début de ce chapitre quelques concepts de base sur le clustering. Puis nous présentons les cartes auto-organisatrices : leur architecture, apprentissage et topologie.

Dans le troisième chapitre nous appliquons la SOM sur un exemple synthétique bidimensionnel afin d'évaluer les performances de ces cartes en fonction de leurs paramètres. Nous appliquons ensuite la SOM sur la base de données : tissus du sein et nous présentons les différents résultats obtenus.

Enfin une conclusion générale conclu ce mémoire.

Chapitre I

Réseau de neurone artificiel

I.1 Introduction :

Les dernières années ont vu un développement technologique puissant dans des domaines divers, et il y a eu un accroissement de besoin pour le contrôle et la gestion des systèmes complexes qui introduisent d'énorme calculs et un nombre de variables important ; d'où la nécessité de chercher de nouvelles méthodes pour une gestion plus souple et moins coûteuse en temps de calculs et en manipulation des variables dont le nombre ne cesse d'augmenter. Pour cela, on s'est intéressé de plus en plus aux systèmes qui apprennent, en utilisant des modélisations des neurones biologiques.

Les réseaux de neurones artificiels ont été étudiés pendant plusieurs années dans le but d'imiter les performances du cerveau de l'être vivant.

Inspirés des réseaux neuromimétique biologiques, ils existent plusieurs modèles de réseaux de neurones artificiels, et chaque modèle se prête bien pour une application particulière (classification, reconnaissance, contrôle ...etc.) ; Mais leurs utilisations restent limitées dans quelques applications, et il reste beaucoup de domaines où les réseaux de neurones n'ont pas trouvé de solutions, telle que la planification par exemple. Les meilleurs systèmes à réseaux de neurones restent assez loin d'imiter des performances telles que celles de l'être humain.

I.2 Définition :

Un réseau de neurones est un modèle mathématique qui tente de reproduire quelques fonctions du cerveau humain, telles que : le parallélisme, l'acquisition des connaissances au travers d'un processus d'apprentissage, le stockage des connaissances et la possibilité d'utilisation de ces connaissances [10].

I.3 Historique :

Dès 1890, W. James, célèbre psychologue américain introduit le concept de mémoire associative et propose ce qui deviendra une loi de fonctionnement pour l'apprentissage sur les réseaux de neurones connue plus tard sous le nom de loi de Hebb.

En 1943 J. Mc Culloch et W. Pitts laissent leurs noms à une modélisation du neurone biologique (un neurone au comportement binaire). Ceux sont les premiers à montrer que des réseaux de neurones formels simples peuvent réaliser des fonctions logiques, arithmétiques et symboliques complexes (tout au moins au niveau théorique).

En 1949 D. Hebb, physiologiste américain explique le conditionnement chez l'animal par les propriétés des neurones eux-mêmes. Ainsi, un conditionnement de type pavlovien tel que, nourrir tous les jours à la même heure un chien, entraîne chez cet animal la sécrétion de salive à cette heure précise même en l'absence de nourriture. La loi de modification des propriétés des connexions entre neurones qu'il propose explique en partie ce type de résultats expérimentaux [3].

En 1957 F. Rosenblatt développe le modèle du Perceptron. Il construit le premier neuro-ordinateur basé sur ce modèle et l'applique au domaine de la reconnaissance de formes.

En 1960 : B. Widrow, un automaticien, développe le modèle Adaline (Adaptative Linear Element). Dans sa structure, le modèle ressemble au Perceptron, cependant la loi d'apprentissage est différente [1].

En 1969, M. Minsky et S. Papert publient un ouvrage qui met en exergue les limitations théoriques du perceptron. Limitations alors connues, notamment concernant l'impossibilité de traiter par ce modèle des problèmes non linéaires. Ils étendent implicitement ces limitations à tous modèles de réseaux de neurones artificiels. Leur objectif est atteint, il y a abandon financier des recherches dans le domaine (surtout aux U.S.A.), les chercheurs se tournent principalement vers les systèmes à bases de règles.

Du 1967 jusqu'à 1982 S. Grossberg, T. Kohonen, Toutes les recherches ne sont pas interrompues Elles se poursuivent sous le couvert de divers domaines comme : le traitement adaptatif du signal, la reconnaissance de formes, la modélisation en neurobiologie, etc [14].

En 1982 : J. J. Hopfield est un physicien reconnu à qui l'on doit le renouveau d'intérêt pour les réseaux de neurones artificiels met en avant l'isomorphisme de son modèle avec le modèle d'Ising (modèle des verres de spins). Cette idée va drainer un flot de physiciens vers les réseaux de neurones artificiels.

En 1983 : La Machine de Boltzmann est le premier modèle connu apte à traiter de manière satisfaisante les limitations recensées dans le cas du perceptron. Mais l'utilisation pratique s'avère difficile, la convergence de l'algorithme étant extrêmement longue.

En 1985 : La rétro-propagation de gradient apparaît nous avons la possibilité de réaliser une fonction non linéaire d'entrée/sortie sur un réseau en décomposant cette fonction en une suite d'étapes linéairement séparables.

I.4 Domaine d'application :

Aujourd'hui, les réseaux de neurones ont de nombreuses applications dans des domaines très variés :

- Traitement d'image : compression d'images, reconnaissance de caractères et de signatures, reconnaissance de formes et de motifs, cryptage, classification, ...
- Traitement du signal : traitement de la parole, identification de sources, filtrage, classification, ...
- Contrôle : diagnostic de pannes, commande de processus, contrôle qualité, robotique, ...
- Optimisation : allocation de ressources, planification, régulation de trafic, gestion, finance, etc. ...
- Simulation : simulation boîte noire, prévisions météorologiques.
- Classification d'espèces animales étant donnée une analyse ADN [10].
- Modélisation de l'apprentissage et perfectionnement des méthodes de l'enseignement.
- Approximation d'une fonction inconnue ou modélisation d'une fonction connue mais complexe à calculer avec précision.

I.5 Principes de modélisation des Réseaux de neurones :

I.5.1 Le neurone biologique :

Les biologistes estiment que le système nerveux compte plus de 100 milliards de neurones interconnectés. Bien que les neurones ne soient pas tous identiques, leur forme et certaines caractéristiques permettent de les répartir en quelques grandes classes. En effet, il est aussi important de savoir que les neurones n'ont pas tous un comportement similaire en fonction de leur position dans le cerveau. Le neurone biologique présenté à la figure I.1 peut être décomposé en quatre principales parties : le corps cellulaire, les dendrites, l'axone et la synapse [2].

a) Dendrites

Chaque neurone possède une chevelure de dendrite qui forme une sorte d'arborescence autour du corps cellulaire, elles permettent aux neurones de capter les signaux qui parviennent de l'extérieur [4].

b) Corps cellulaire

Dans la plupart des cas, la forme du corps cellulaire dépend de sa position dans le cerveau, elle peut être pyramidale ou sphérique. Il inclut le noyau du neurone et effectue les transformations biochimiques nécessaires à la vie du neurone [4].

c) L'axone

L'axone est la fibre nerveuse qui permet la transposition des signaux émis par le neurone généralement un axone est plus long que les dendrites, ainsi se ramifie à son extrémité, là où il communique avec les autres neurones [4].

d) Les synapses

Ce sont des jonctions entre deux neurones et qui sont essentielles dans le fonctionnement du système nerveux [4].

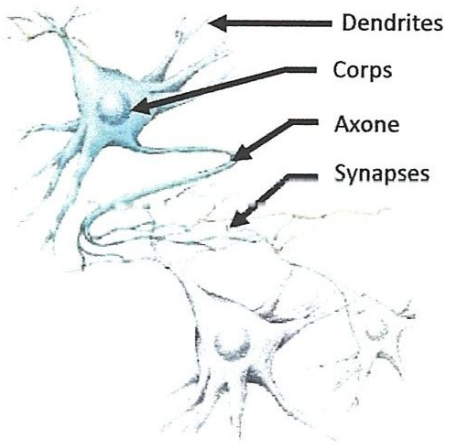


Figure I.1 : Schéma d'un neurone biologique

I.5.2 Fonctionnement de neurone biologique :

Les dendrites forment un maillage de récepteurs nerveux qui permettent d'acheminer vers le corps du neurone, des signaux électriques en provenance d'autres neurones. Celui-ci agit comme une espèce d'intégrateur en accumulant des charges électriques lorsque le neurone devient suffisamment excité, par un processus électrochimique, il engendre un potentiel électrique qui se propage jusqu'à trouver son axone pour éventuellement venir exciter d'autres neurones. Le point de contact entre l'axiome d'un neurone et la dendrite d'un autre neurone s'appelle la synapse. Il semble que c'est l'arrangement spatial des neurones et de leur axone, ainsi que la qualité des connexions synaptiques individuelles qui détermine la fonction précise d'un réseau de neurones biologique [11].

• Le schéma classique présenté par les biologistes (Figure I.2) est celui d'un somma qui exécute une sommation de tous les signaux transmis par ses dendrites et envoie un influx nerveux à son axone ; si la sommation dépasse un certain seuil

•L'évolution des connections entre les neurones permet la mémorisation et l'apprentissage tel que :

La mémorisation : sert à l'appel des propriétés acquises par l'apprentissage afin d'être utilisé aux modifications.

L'apprentissage : est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré.

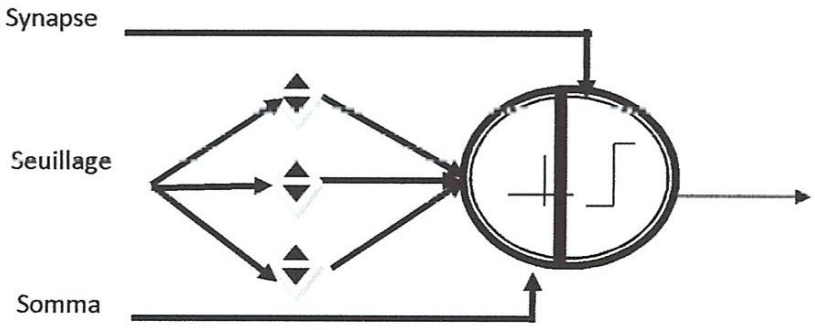


Figure I.2 : Le schéma classique présenté par les biologistes

I.5.3 Le neurone artificiel :

Un neurone artificiel est une fonction mathématique conçu comme un modèle de neurones biologiques. Les neurones artificiels sont des unités constitutives dans un réseau neuronal artificiel. Selon le modèle utilisé, ils peuvent être appelés une unité semi-linéaire, Nv neurone, neurone binaire, la fonction de seuil linéaire, ou McCulloch-Pitts (MCP) neurone.

Le neurone artificiel reçoit une ou plusieurs entrées (dendrites représentant) et les résume pour produire une sortie (représentant de l'axone d'un neurone). Habituellement, les valeurs de chaque nœud sont pondérées, et la somme est passée à travers une fonction non-linéaire connue en tant que fonction de l'activation ou de la fonction de transfert. Les fonctions de transfert ont généralement une forme sigmoïde, mais ils peuvent aussi prendre la forme d'autres fonctions non-linéaires, des fonctions linéaires par morceaux, ou fonctions en escalier. Ils sont aussi souvent monotones croissante, continue, dérivable et bornée [9].

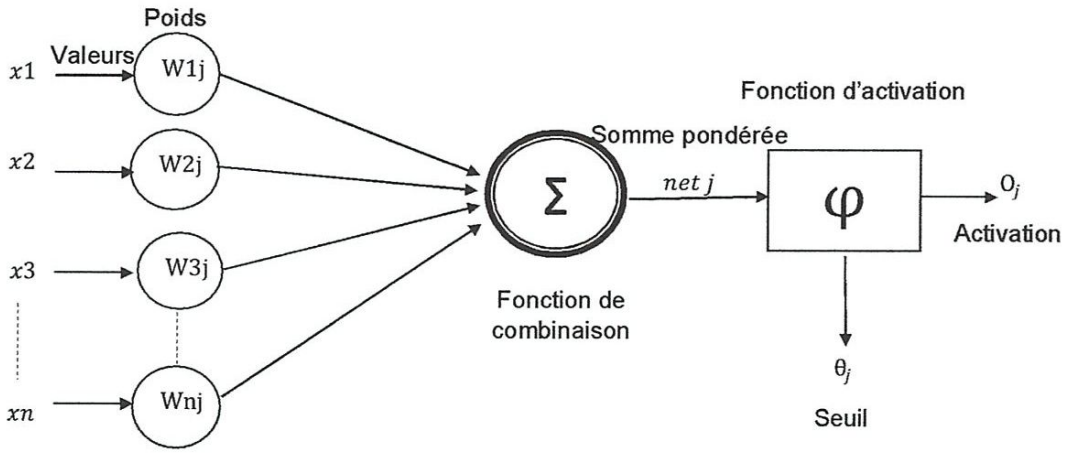


Figure I.3 : Structure générale d'un neurone artificiel

I.5.4 Fonction d'activation

Il est clair que la fonction d'activation joue un rôle très important dans le comportement du neurone. Elle retourne une valeur représentative de l'activation du neurone, cette fonction a comme paramètre la somme pondérée des entrés ainsi que le seuil d'activation La nature de cette fonction diffèrent selon le réseau. On en compte divers types, parmi elles :

- Fonction binaire à seuil

Fonction à seuil montrée par la figure I.4(a) et définie par :

$$h(x) = \begin{cases} 1 & \text{si} \quad x > 0 \\ 0 & \text{sinon} \end{cases} \qquad (I.1)$$

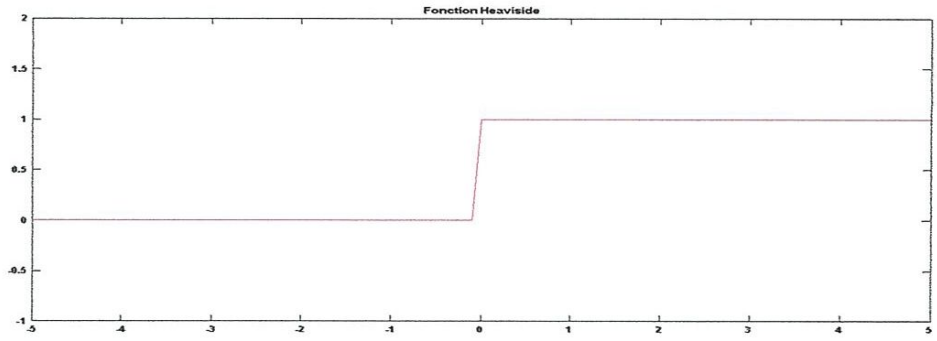


Figure I.4(a) : Fonction binaire à seuil

Fonction signe montrée par la figure I.4(b) et définie par :

$$sgn(x) = \begin{cases} 1 & \text{si} \quad x > 0 \\ -1 & \text{sinon} \end{cases} \tag{I.2}$$

Le seuil introduit une non-linéarité dans le comportement du neurone, cependant, il limite la gamme des réponses possibles a deux valeurs.

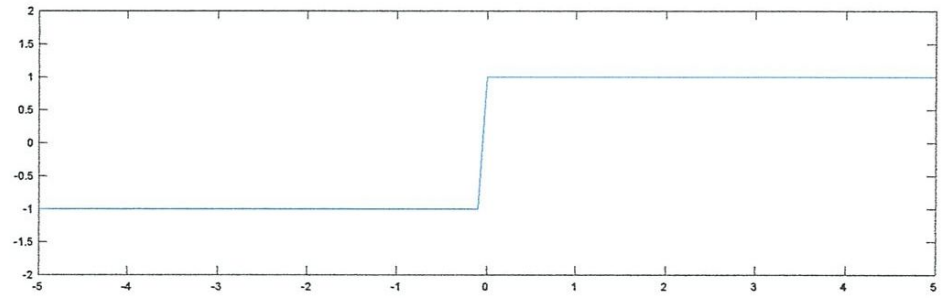


Figure I.4(b) : Fonction signe

• Fonction linéaire

C'est l'une des fonctions d'activations les plus simples, cette fonction est représenterpar la figure I.4(c), sa fonction est définie par :

$$F(x) = x \tag{I.3}$$

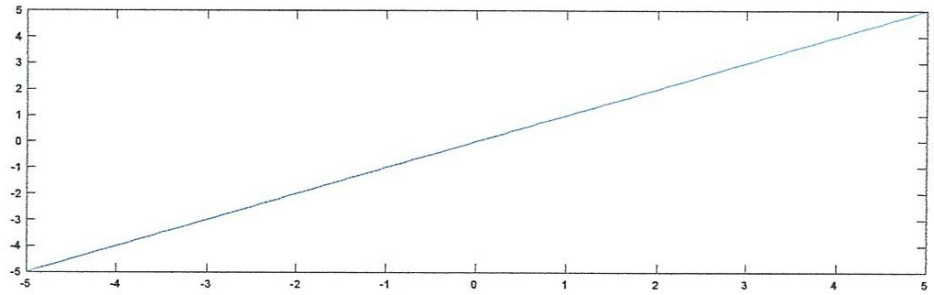


Figure I.4(c) : Fonction lineaire

• Fonction linéaire à seuil ou multi-seuils

On peut la définir comme suit :

$$h(x) = \begin{cases} x & \text{si} \quad x \in [u, v] \\ v & \text{si} \quad x < v \\ u & \text{si} \quad x > v \end{cases} \tag{I.4}$$

Cette fonction représente un compromis entre la fonction linéaire et la fonction seuil. Son graphe est représenté par la figure I.4(d) : entre ses deux barres de saturation, elle confère au neurone une gamme de réponses possibles. En modulant la pente de la linéarité, on affecte la plage de réponse du neurone.

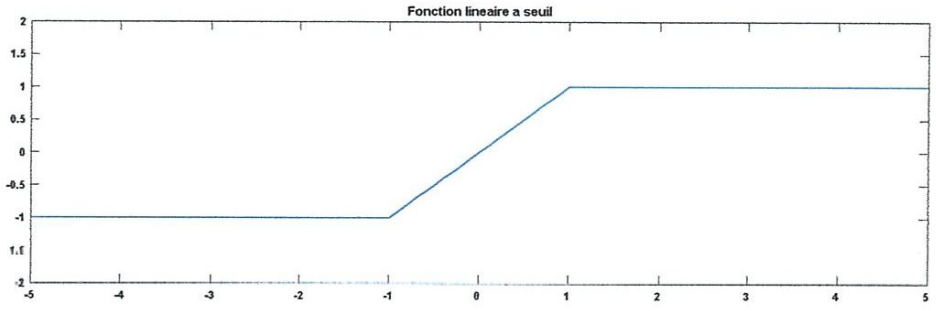


Figure I.4(d) : Fonction linéaire a seuil

• Fonction sigmoïdale

Elle est l'équivalent continu de la fonction linéaire représentée par la figure I.4(e). Etant continue, elle est dérivable, d'autant plus que sa dérivée est simple à calculer.

La fonction log sigmoïde est définie par :

$$logsig(x) = \frac{1}{1+e^{-x}} \tag{I.5.a}$$

Et sa dérivée est :

$$\frac{d}{dx}\big(logsig(x)\big) = logsig(x)\big(1 - logsig(x)\big) \qquad \text{(I.5.b)}$$

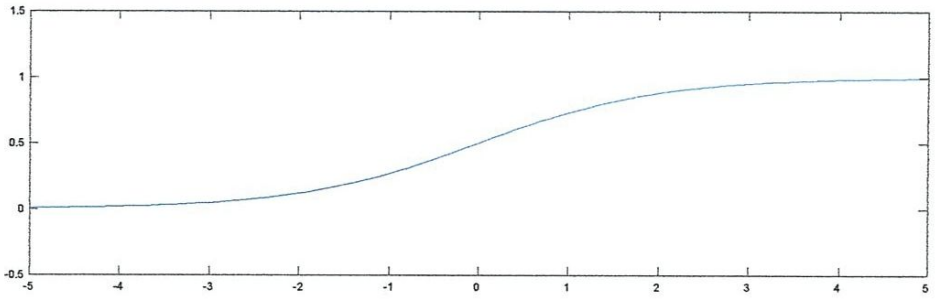


Figure I.4(e) : Fonction log sigmoïde

• Fonction tangente sigmoïde

La fonction tangente sigmoïde est celle montrée par la figure I.4(f). Elle est définie par :

$$tansig(x) = \frac{2}{(1+e^{-2})} - 1 \qquad \text{(I.6.a)}$$

Et sa dérivée est définie comme suit :

$$\frac{d}{dx}(tansig(x)) = \frac{4e^{-2x}}{(e^{-2x}+1)^2} \qquad \text{(I.6.b)}$$

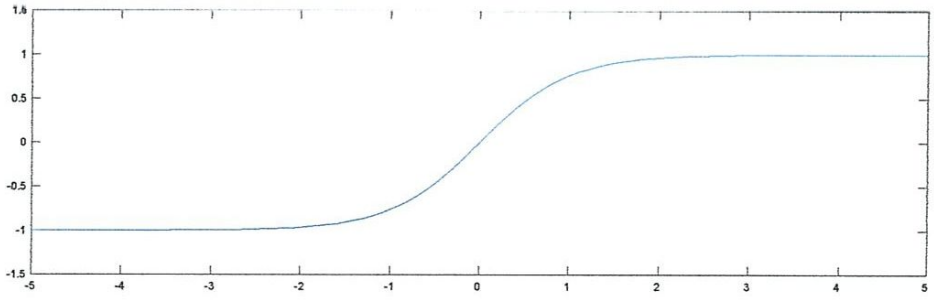


Figure I.4(f) : Fonction tangente sigmoïde

I.6 Apprentissage d'un réseau de neurones :

I.6.1 Définition :

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré. L'apprentissage neuronal fait appel à des exemples d'apprentissage.

Dans les algorithmes actuels, les variables modifiées pendant l'apprentissage sont les poids des connexions [10].

I.6.2 Protocoles d'apprentissages :

Presque la totalité des réseaux de neurones ont en commun un même protocole d'apprentissage celui-ci comporte quatre étapes :

Etape 1 : Initialisation des poids synaptiques avec des petites valeurs aléatoires.

Etape 2 : Présentation de l'exemple d'entré et propagation de l'activation des neurones.

Etape 3 : Calcule de l'erreur. Dans le cas d'un apprentissage supervisé cette erreur dépend de la différence entre l'activation des neurones et la sortie désirée.

Etape 4 : Calcul du vecteur de correction à partir des valeurs des erreurs, avec lequel on effectue la correction des poids synaptiques.

I.6.3 Les types d'apprentissage :

Les techniques d'apprentissage se subdivisent en trois grandes familles :

I.6.3.1 Apprentissage supervisé :

Pour ce type d'apprentissage (perceptron, Adaline, etc...), le réseau doit savoir qu'il a commis une erreur et il doit connaitre la réponse qu'il dû donner. Pour un stimulus présenté à la couche d'entrée du réseau, la réponse obtenus est comparée avec celle désirée et la modification et les ajustements à apporter aux poids sont déterminés en fonction de l'erreur commise par le réseau. Généralement, les règles d'apprentissage supervisé sont des formes de descente du gradient.

L'apprentissage supervisé nécessite donc la définition d'une base d'exemples d'apprentissage représentative. Chaque exemple présenté au réseau est un couple (entrée, sortie désirée). La minimisation de l'erreur entre la valeur de sortie et la valeur désirée est basée sur le principe de l'erreur quadratique [2].

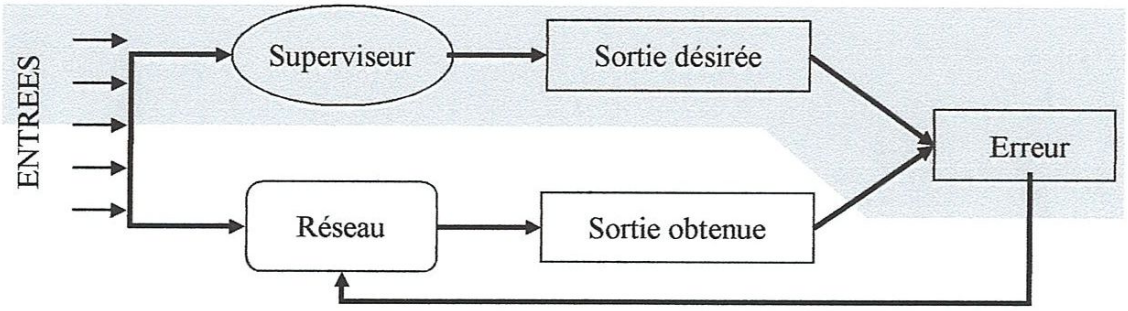


Figure I.5(a) : Apprentissage supervise

I.6.3.2Apprentissage semi-supervisé :

L'apprentissage semi-supervisé suppose qu'un comportement de référence précis n'est pas disponible, mais qu'en revanche, il est possible d'obtenir des indications qualitatives (correcte /incorrecte) sur les performances du réseau [2].

I.6.3.3 Apprentissage non supervisé :

Pour ce type d'apprentissage il s'agit d'atteindre l'ensemble des poids synaptiques pour lesquels le comportement du réseau est optimal. La modification et l'ajustement des poids se font en fonction d'un critère interne, indépendant de la relation entre le comportement du réseau et la tâche qui doit effectuer.

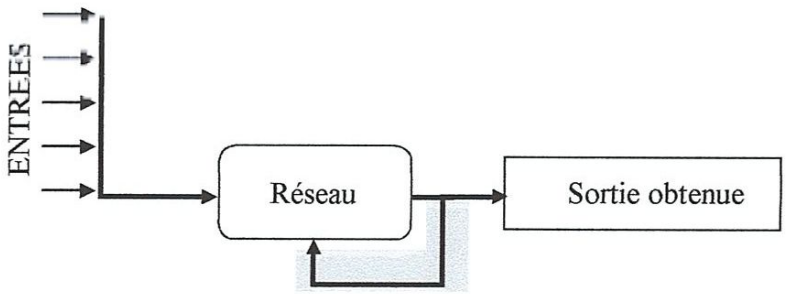


Figure I.5(b) : Apprentissage non supervisé

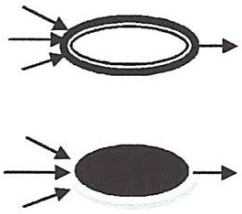
I.6.4 Règles d'apprentissage :

I.6.4.1 La règle de Hebb :

La règle de Hebb est le premier mécanisme d'évolution proposé sur les synapses. Son interprétation pour les réseaux de neurones artificiels est la suivante :

On considère que deux neurones connectés entre eux sont activé aux mêmes moments, la connexion qui les relie doit être renforcée et elle n'est pas modifié, dans le cas contraire. C'est -à-dire que le poids Wij d'une connexion entre un neurone i et un neurone j augmente quand les deux neurones sont activés en même temps et il n'est pas modifié, dans le cas contraire [2], [3].

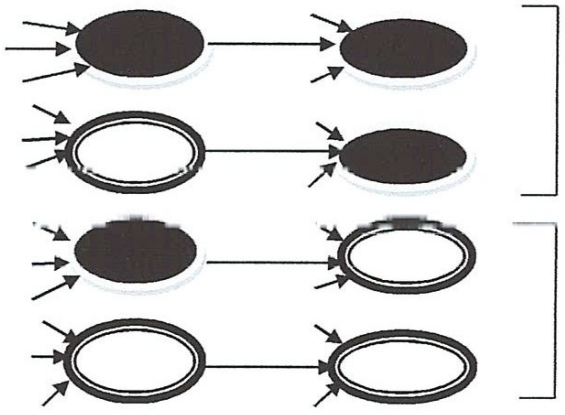
Si nous prenons, à titre d'exemple, les conventions suivantes :



Cellule activée

Cellule non activée

La règle de Hebb donne alors :



Connexion renforcée

Connexions non renforcées

Ceci se traduit par :

Quand la cellule émettrice et la cellule réceptrice s'activent en même temps, il faut augmenter le poids de cette connexion lors de l'apprentissage. La connexion entre ces deux cellules devienne alors très forte. Si la cellule émettrice s'active sans que la cellule réceptrice ne le

soit, ou si la cellule réceptrice s'active alors que la cellule émettrice n'était pas activée, cela traduit bien le fait que la connexion entre les deux n'est pas prépondérante dans le comportement de la cellule réceptrice on peut donc dans la phase d'apprentissage laisser un poids faible à cette connexion [3].

En se basant sur ce principe, Hebb a donné la règle d'apprentissage suivante :

$$Wij(t + \partial t) = Wij(t) + \mu AiAj \quad \text{(I.7)}$$

Avec Wij (t) et $Wij(t + \partial t)$: les poids de la connexion entre le neurone i et le neurone j aux instants(t)et$(t + \partial t)$.

Ai Et Aj : L'activation du neurone i et l'activation du neurone j

μ : C'est le paramètre de l'intensité de l'apprentissage$(\mu > 0)$.

I.6.4.2 La règle de Widrow-Hoff :

La règle d'apprentissage de Widrow-Hoff, ou des moindres carrés (LMS, Least Square Sum), est une règle d'apprentissage supervisé basée sur la correction d'erreurs observées en sortie. Cette règle consiste à minimiser une fonction coût caractérisée par l'erreur quadratique moyenne. Pour un ensemble d'apprentissage contenant Qpaires entrée/sortie désirée$\{(X^{(q)}/T^{(q)})\}, q = 1, \dots Q$ où $X^{(q)}$ et $T^{(q)}$ représentent respectivement la $q^{ème}$ entrée et la $q^{ème}$ sortie désirée, l'erreur $(e(r))$ à l'itération r est donnée par :

$$e(r) = T(r) - Y(r) \quad \text{(I.8)}$$

Où :$Y(r)$ est la sortie calculée du réseau. La fonction coût est :

$$F(X) = e^2(r) \quad \text{(I.9)}$$

L'apprentissage selon la règle LMS consiste à calculer le gradient à chaque présentation d'un exemple d'apprentissage. Le changement de poids est alors :

$$\Delta w_{ij}(t) = -\eta \nabla F(X) \quad \text{(I.10.a)}$$

$$= \eta \frac{\partial e^2(r)}{\partial w_{ij}} \quad \text{(I.10.b)}$$

Cette règle de correction permet donc aux neurones d'adapter leurs poids pour se rapprocher à une valeur désirée correspondante à chaque exemple présenté. Cette règle a été utilisée pour l'apprentissage de l'ADALINE dans lequel chaque neurone i corrige ses poids w_{ij} à l'itération r selon l'équation suivante :

$$\Delta w_{ij}(r) = \Delta w_{ij}(r-1) - \eta(t_i - y_i)x \qquad (I.11)$$

Où : t_i et $y_i$sont respectivement la sortie désirée et la sortie calculée correspondantes au neurone i ; x est l'entrée et η est une constante positive appelée pas d'apprentissage [10].

I.7 Architecture des réseaux de neurone :

On distingue deux grands types d'architectures de réseaux de neurones : les réseaux de neurones non bouclés et les réseaux de neurones bouclés.

I.7.1 Les réseaux de neurones non bouclés :

Un réseau de neurones non bouclé réalise une (ou plusieurs) fonctions algébriques de ses entrées, par composition des fonctions réalisées par chacun de ses neurones.

La Figure I.6(a) représente un réseau de neurones non bouclé qui a une structure particulière, très fréquemment utilisée : il comprend des entrées, une couche de neurones "cachés" et des neurones de sortie. Les neurones de la couche cachée ne sont pas connectés entre eux. Cette structure est appelée Perceptron multicouche.

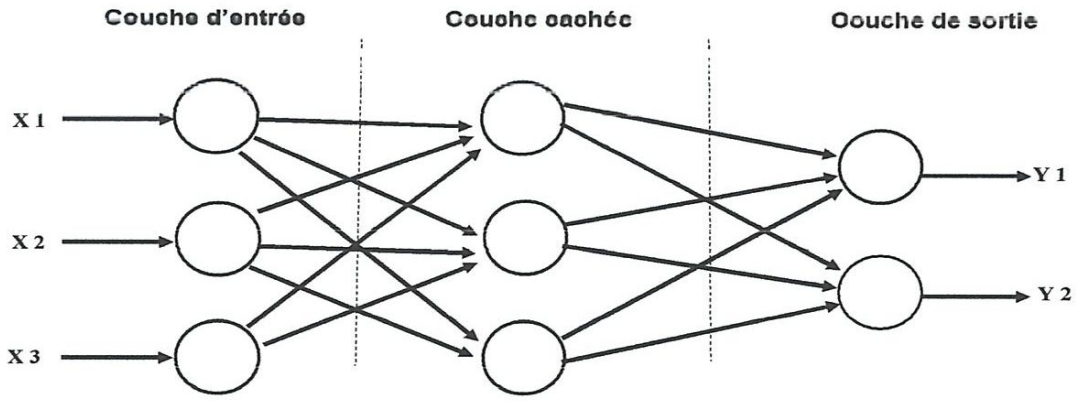


Figure I.6(a) : Réseaux de neurones non bouclés

I.7.2 Les réseaux de neurones bouclés :

Les réseaux de neurones bouclés peuvent avoir une topologie de connections quelle conque, comprenant notamment des boucles qui ramènent aux entrées la valeur d'une ou plusieurs sorties.

La figure I.6(b) montre un exemple d'un réseau de neurone bouclé.

Pour chaque système soit causal, il faut évidemment qu'a toute boucle soit associe un retard. Un réseau de neurone bouclé à temps discret est donc régi par une ou plusieurs équations différentielles non linéaires, résultant de la composition des fonctions réalisées par chacun des neurones et des retards associés à chacune des connexions.

La forme la plus générale des équations régissant un réseau de neurones bouclé et appelée forme canonique :

$$x(k+1) = \varphi[x(k), u(k)] \quad \text{(I.12)}$$

$$y(k) = \psi[x(k), u(k)] \quad \text{(I.13)}$$

Où $x(k)$est le vecteur des variables d'état à l'instant (discret) kT, $u(k)$ est le vecteur des entrées, $y(k)$ est le vecteur des sorties.

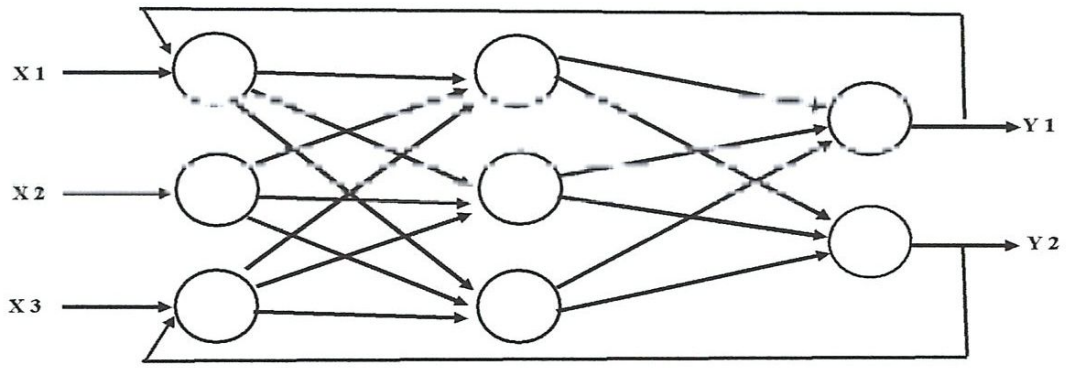


Figure I.6(b) : Réseaux de neurones bouclés

I.8 Avantages et Inconvénients des réseaux de neurones :

I.8.1 Avantages des réseaux de neurones :

Capacité de représenter n'importe quelle fonction, linéaire ou pas, simple ou complexe.

Faculté d'apprentissage à partir d'exemples représentatifs,

Résistance au bruit ou au manque de fiabilité des données.

Simple à manipuler, beaucoup moins de travail personnel à fournir que dans l'analyse statistique classique. Ne demande pas de grandes compétences en mathématique, informatique ou statistique.

Comportement moins mauvais en cas de faible quantité de données.

Pour l'utilisateur novice, l'idée d'apprentissage est plus simple à comprendre que les complexités des statistiques multi variables.

I.8.2 inconvénients des réseaux de neurones :

L'absence de méthode systématique permettant de définir la meilleure topologie du réseau et le nombre de neurones à placer dans la (ou les) couche(s) cachée(s).

Le choix des valeurs initiales des poids du réseau et le réglage du pas d'apprentissage, qui jouent un rôle important dans la vitesse de convergence.

Le problème du sur apprentissage (apprentissage au détriment de la généralisation).

La connaissance acquise par un réseau de neurone est codée par les valeurs des poids sont incompréhensibles pour l'utilisateur.

I.9 Conclusion :

Les réseaux de neurones ont connu un essor considérable tant qu'en nouvelles architectures qu'en nouveaux algorithmes d'apprentissage, et dans ce chapitre nous avons tenté de donner un simple survol sur ces importants outils mathématiques.

Chapitre II

Clustering par SOM

II.1 Généralités sur le Clustering :

II.1.1 Définition :

Le clustering, ou regroupement, peut être considéré comme le problème le plus important de l'apprentissage non supervisé ; ainsi, comme tous les autres problèmes de ce genre, il traite de la recherche d'une structure dans un ensemble de données non marquées (qu'on ignore leurs appartenances) [8].

Une définition simple du clustering pourrait être : c'est le processus d'organisation des objets dans des clusters dont les membres sont similaires d'une certaine façon.

Un cluster est donc une collection d'objets qui sont « similaires » entre eux et sont "dissemblables" pour les objets appartenant à d'autres groupes. Ceci peut être montré avec un exemple graphique simple (figure II.1). Dans cet exemple, nous identifions facilement les 4 clusters dans lequel les données peuvent être divisées, le critère de similarité est distance : deux ou plusieurs objets appartiennent au même groupe si elles sont « proches », selon une distance donnée (dans ce cas à distance géométrique). Ceci est appelé cluster basé sur la distance.

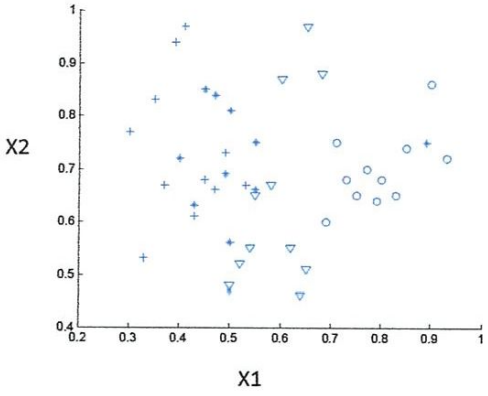


Figure II.1 (a) : Clusters difficile a traité

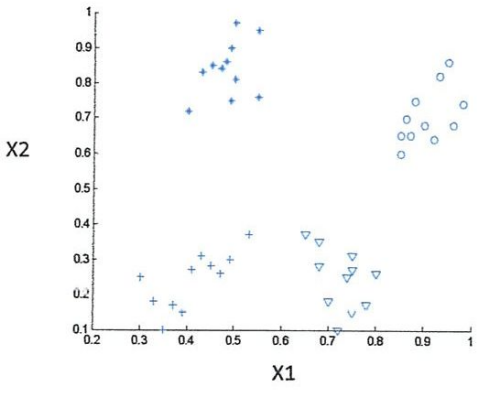


Figure II.1 (b) : Clusters facile a traité

Un autre type de regroupement est le regroupement conceptuel : deux ou plusieurs objets appartiennent au même groupe si celui-ci définit un concept commun à tous que les objets. En d'autres termes, les objets sont regroupés en fonction de leur adéquation à des concepts descriptifs et non pas en fonction de mesures de similarité simples.

II.1.2 Les objectifs du Clustering :

Le clustering a pour objectif de déterminer le groupement intrinsèque dans un ensemble de données non marquées. Mais comment décider ce qui constitue un bon regroupement ? En effet, il n'y a pas de "meilleure" critère absolu qui pourrait être de l'objectif final de tout processus de clustering. Par conséquent, c'est l'utilisateur qui doit fournir un critère d'une manière telle que le résultat de ce processus répond aux besoins de son application.

Par exemple, nous pourrions être intéressés à trouver des représentants pour les groupes homogènes (réduction des données), dans la recherche de "clusters naturelles" et décrire leurs propriétés inconnues (types de données « naturelles »), dans la recherche de groupements utiles et appropriés (classes de données "utiles") ou trouver des objets de données inhabituelles (détection des valeurs aberrantes) [8].

II.1.3 Applications du clustering :

Les algorithmes de clustering peuvent être appliqués dans de nombreux domaines, par exemple :

• Marketing : trouver des groupes de clients avec un comportement similaire donné dans une grande base de données des clients contenant leurs propriétés et les dossiers d'achat antérieurs.

• Biologie : classification des plantes et des animaux compte tenu de leurs caractéristiques.

• Bibliothèques : commande de livres.

• Assurance : l'identification des groupes de titulaires de polices d'assurance automobile avec un coût moyen des sinistres élevés ; identifier les fraudes.

• Ville de planification : l'identification des groupes de maisons en fonction de leur type de maison, la valeur et l'emplacement géographique.

• Les études sismiques : formation de cartes des tremblements de terre pour identifier les zones dangereuses.

• Classification des documents : le regroupement des données des documents dans le web pour découvrir des groupes de modèles d'accès similaires.

II.1.4 Mesure de similarité :

Du fait que la similarité est fondamentale pour la définition d'un cluster, une mesure de la similitude entre deux exemples du même espace caractéristique est essentielle pour la plupart des procédures de clustering. En raison de la variété des types et des échelles des caractéristiques, la mesure de distance doit être choisie avec soin. Il est plus fréquent de calculer la dissemblance entre deux exemples en utilisant une mesure de distance définie sur l'espace caractéristique [13]. Nous allons nous concentrer sur les mesures de distance les plus utilisés pour les modèles dont les caractéristiques sont continuées. La mesure la plus populaire pour les fonctions continues est la distance euclidienne donnée par :

$$d_2(x_i, x_j) = \left(\sum_{k=1}^{d} (x_i, k - x_j, k)^2\right)^{\frac{1}{2}} = \|x_i - x_j\|^2 \qquad \text{(II.1)}$$

Cette fonction est un cas particulier $(p = 2)$ de la fonction Minkowski, donnée par :

$$d_p\ (x_i, x_j) = \left(\sum_{k=1}^{d} |x_i, k - x_j|^{\mathrm{P}}\right)^{\frac{1}{p}} = \|x_i - x_j\|^{\mathrm{P}} \qquad \text{(II.2)}$$

La distance euclidienne est intuitive et elle est communément utilisée pour évaluer la proximité des objets dans l'espace à deux ou trois dimensions. L'inconvénient de l'utilisation directe des paramètres Minkowski est que la plus grande échelle pourrait dominer les autres. Les solutions à ce problème comprennent la normalisation des caractéristiques. La corrélation linéaire entre les caractéristiques peut aussi fausser les mesures de distance ; cette déformation peut être atténuée en appliquant une transformation des données à l'aide de la distance de Mahalanobis donnée par :

$$d(x_i, x_j) = (x_i - x_j)F^{-1}(x_i - x_j)^{\mathrm{T}} \qquad \text{(II.3)}$$

Ou F est la matrice de covariance

II.2 La carte auto-organisatrice de Kohonen :

II. 2. 1 Introduction :

Une Carte Auto-Organisatrice (SOM, Self-Organizing Map) se compose d'un ensemble de neurones artificiels, qui représentent la structure des données. Les neurones sont connectés avec des connexions topologiques pour former une grille à deux dimensions. Deux neurones proches devraient représenter des données similaires, deux neurones distants (sur la carte)

doivent représenter des données différentes. Ces propriétés sont assurées pendant le processus d'apprentissage grâce aux informations de voisinage qui imposent des contraintes topologiques. Toutefois, dans l'algorithme SOM, l'information topologique est fixée avant le processus d'apprentissage et peut ne pas être pertinente par rapport à la structure des données. Pour résoudre ce problème, certains travaux ont été réalisés afin d'adapter le nombre de neurones au cours du processus d'apprentissage en fonction des données à analyser. Les résultats ont montré que la qualité du modèle est améliorée lorsque le nombre de neurones est appris à partir des données. En dépit de ces résultats, il y a très peu de travaux qui abordent le problème de l'apprentissage des contraintes topologiques en fonction de la structure des données. Pourtant, à la fin du processus d'apprentissage, des neurones "voisins" peuvent ne pas représenter des données similaires [14].

II. 2. 2 Principe de fonctionnement :

Les cartes auto-organisatrices ont été introduites par T. Kohonen en 1981 en s'inspirant du fonctionnement des cartes topographiques du cerveau humain, tel que, les points proches qui se trouvent dans le corps humain sont représentés par des groupes de neurones proches dans le cerveau. Ces cartes ne sont pas uniformes, à savoir, la surface la plus sensible du corps humain est représentée par une zone contenant le plus grand nombre de neurones. D'un point de vue informatique, on peut traduire cette propriété de la façon suivante : supposons que l'on dispose d'un ensemble de données que l'on désire classifier. On cherche un mode de représentation tel que les objets voisins soient classés dans la même classe ou dans des classes voisines. Ce type de réseaux de neurones artificiels a largement montré son efficacité dans la classification de données multidimensionnelles, mais malheureusement il a été ignoré pour de nombreuses années malgré son grand intérêt. Le principe des cartes de Kohonen est de projeter un ensemble complexe de données sur un espace de dimension réduite (2 ou3). Cette projection permet d'extraire un ensemble de vecteurs dites référents ou prototypes. Ces prototypes sont caractérisés par des relations géométriques simples. La projection de données par SOM se produit tout en conservant la topologie et les métriques les plus importantes des données d'entrée lors de l'affichage, c'est-à-dire les données proches (dans l'espace d'entrée) vont avoir des représentations proches dans l'espace de sortie et vont donc être classés dans le même cluster ou dans des clusters [14].

II. 3 L'algorithme d'apprentissage de la SOM

II. 3. 1 Principe :

Après l'initialisation des valeurs de chaque neurone on présente une à une les données d'apprentissage à la carte auto organisatrice. Selon les valeurs des neurones, il y en a un qui répondra le mieux. Celui dont la valeur sera la plus proche de la donnée présentée. Alors ce neurone sera adapté avec un changement de valeur pour qu'il réponde encore mieux à un autre stimulus de même nature que le précédent. De même, on adapte aussi les neurones voisins du gagnant avec un facteur multiplicatif du gain inférieur à un. Ainsi, c'est toute la région de la carte autour du neurone gagnant qui se spécialise. En fin d'algorithme, lorsque les neurones ne bougent plus, ou très peu, à chaque itération, la carte auto organisatrice recouvre toute la topologie des données. C'est toute la région de la carte autour du neurone gagnant qui se spécialise.

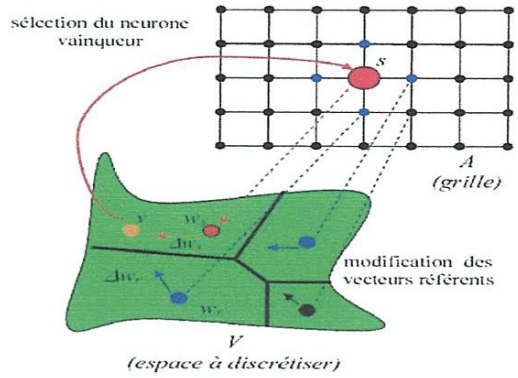


Figure II.2 : Principe d'apprentissage de la SOM

La figure II.2 donne une représentation de l'algorithme d'auto-organisation pour le modèle de Kohonen. Chaque neurone a un vecteur référent qui le représente dans l'espace d'entrée. Lorsque qu'un vecteur d'entrée v est présenté, le neurone vainqueur s est sélectionné, le plus proche dans l'espace d'entrée. Le vecteur référent du vainqueur ws est rapproché de v. Les vecteurs référents des autres neurones sont aussi déplacés vers v, mais avec une amplitude moins importante.

II. 3. 2 Notion de voisinage :

Tout comme dans le cortex, les neurones sont reliés les uns aux autres, c'est la topologie de la carte. La forme de la carte définie les voisinages des neurones et donc les liaisons entre neurones. La forme de la carte définit les voisinages des neurones et donc les liaisons entre neurones. La fonction de voisinage décrit comment les neurones dans la proximité du vainqueur *s* sont entraînés dans le mouvement de correction. On utilise en général :

$$\mathrm{NS}\,(\mathrm{r},\mathrm{s},\mathrm{t}) = \exp\left(\frac{\|\vec{r}-\vec{s}\|}{2\sigma^2(t)}\right) \qquad \text{(II.4)}$$

Où σ s'appelle coefficient de voisinage. Son rôle est de déterminer un rayon de voisinage autour du neurone vainqueur. La fonction de voisinage NS force les neurones qui se trouvent dans le voisinage de *s* à rapprocher leurs vecteurs référents du vecteur d'entrée *v*. Moins un neurone est proche du vainqueur dans la grille, moins son déplacement est important. La correction de vecteurs référents est pondérée par les distances dans la grille. Cela fait apparaître, dans l'espace d'entrée, les relations d'ordre dans la grille [12].

Pendant l'apprentissage la carte décrite par les vecteurs référents du réseau évolue d'un état aléatoire vers un état de stabilité dans lequel elle décrit la topologie de l'espace d'entrée tout en respectant les relations d'ordre dans la grille.

La SOM permet donc :

- Similitude des densités dans l'espace d'entrée :

La carte reflète la distribution des points dans l'espace d'entrée. Les zones dans lesquelles les vecteurs d'entraînement *v* sont tirés avec une grande probabilité d'occurrence sont cartographiées avec une meilleure résolution que les zones dans lesquelles les vecteurs d'entraînement *v* sont tirés avec une petite probabilité d'occurrence.

- Préservation des relations topologiques :

Des neurones voisins dans la grille occupent des positions voisines dans l'espace d'entrée (préservation des voisinages de la grille), et des points proches dans l'espace d'entrée se projettent sur des neurones voisins dans la grille (préservation de la topologie de l'espace d'entrée). Les neurones ont tendance à discrétiser l'espace de façon ordonnée.

II. 3. 3 Fonction de voisinage :

Avec la fonction de voisinage, tous les vecteurs voisins sont décalés vers le vecteur d'entrée, Cependant, la mise à jour de neurone gagnant est le plus importante et plus loin un neurone voisin, moins son poids est modifié. La fonction NS détermine comment le réglage du poids s'intègre avec la distance du vainqueur. Il existe plusieurs possibilités pour cette fonction et les plus couramment utilisées sont les fonctions linéaire, gaussien et exponentielle [12].

La forme la plus simple de la fonction NS est la fonction décroissance linéaire, où la puissance diminue de manière linéaire avec la distance du neurone gagnant. La forme gaussienne de la fonction NS rend l'ajustement des poids en douceur avec la distance, comme le montre la figure II.3.

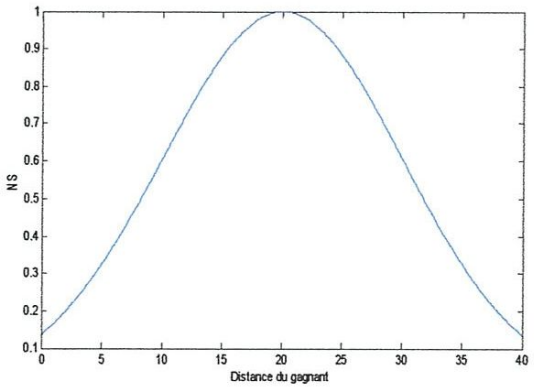


Figure II.3 : La forme gaussienne de la fonction NS

La fonction gaussienne est donnée par :

$$\mathrm{N(j)} = \exp\left(\frac{-\mathrm{d}^2(\mathrm{i,j})}{2\sigma^2}\right) \qquad \text{(II.5)}$$

Où $d(i,j)$ est la distance entre le neurone gagnant i et toute autre neurone j et s est la largeur de la gaussienne. Cette largeur est généralement définie en fonction du rayon de la région avoisinante, et la largeur de la fonction illustrée à la figure II.3 est 20. La valeur est maximale $(1,0)$ au neurone gagnant, qui est positionné au centre de la figure II.3.

La fonction exponentielle de décroissance NS est donnée par :

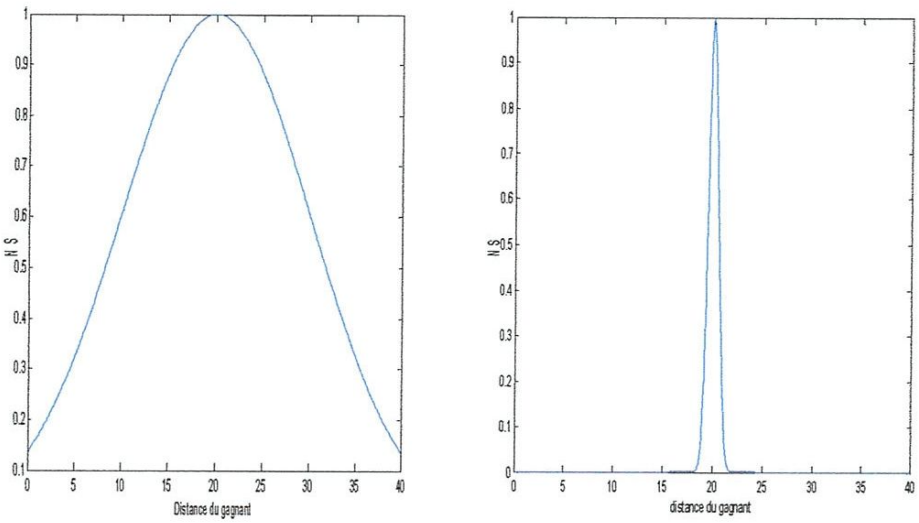
$$N(j) = \exp(-kd^2(i,j)) \qquad \text{(II.6)}$$

Où k est une constante.

II. 3. 4 Les deux phases d'apprentissage de la SOM :

L'apprentissage est habituellement effectué en deux phases : la phase de compétition et la phase d'adaptation. Dans la phase de compétition, le pas d'apprentissage et la taille de voisinage sont réduits avec les itérations jusqu'à ce que le gagnant ou quelques voisins autour du gagnant restent (figure II.5). Dans cette phase, l'ordre topologique des vecteurs de poids se produit. Selon les itérations requises, une attention particulière doit être accordée au choix du pas d'apprentissage et la fonction de voisinage. Une recommandation est que le paramètre du pas d'apprentissage doit commencer par une valeur relativement élevé et devrait ensuite diminuer progressivement, mais doit rester au-dessus de $0{,}01$. La taille de voisinage doit couvrir initialement presque tous les neurones dans le réseau lorsqu'elle centré sur un neurone gagnant puis rétrécir lentement avec les itérations. Selon le problème, la phase de compétition peut prendre quelques à des milliers d'itérations, au cours de laquelle, la taille de voisinage est autorisée de réduire à quelques neurones autour des neurones gagnants ou tout simplement le gagnant lui-même.

Dans la phase de d'adaptation, la carte est affinée avec la réduction du voisinage de façon à produire une représentation précise de l'espace d'entrée. Cette phase peut également exécuter à partir de quelques centaines ou des milliers d'itérations. Dans cette phase, le pas d'apprentissage est maintenu à une faible valeur, sur l'ordre de $0{,}01$, pour parvenir à une convergence avec une bonne précision statistique. Le pas d'apprentissage se réduit d'une façon exponentielle pour empêche la valeur zéro, permettant ainsi à la carte de convergé lentement. Avec décroissance linéaire du pas d'apprentissage la fonction NS doit contenir seuls les voisins les plus proches du neurone gagnant et peut réduire lentement un ou zéro voisin (à savoir, seul le vainqueur reste) [13].



(a) **(b)**

Figure II. 5 : Fonction de voisinage, (a) ou début d'apprentissage, (b) fin d'apprentissage

Réduction du pas d'apprentissage *:*

Au cours de l'apprentissage le pas d'apprentissage β, est réduit avec des itérations et une forme courante de cette fonction est la décroissance linéaire, donné par :

$$\beta(\iota) = \beta_0 (1 - \iota / T) \qquad \text{(II.7)}$$

Où β_0 et $\beta(t)$ sont respectivement le taux d'apprentissage initial et pas d'apprentissage à l'itération t. T Est une constante qui permet de régler la réduction du pas d'apprentissage. Une autre forme est la décroissance exponentielle du taux d'apprentissage donnée par :

$$\beta(t) = \beta_0 \, exp\,(-t/T) \qquad \text{(II.8)}$$

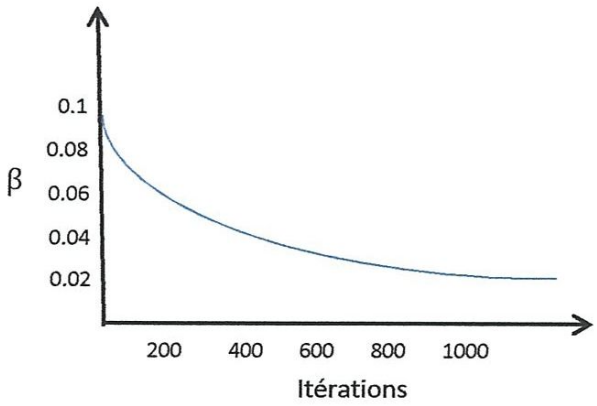


Figure II.6 : La dégradation de pas d'apprentissage avec les itérations

II. 4 Topologie de la SOM :

Il existe plusieurs façons de définir un voisinage. Linéaire, carré, hexagonaux, la figure II.7 représente les topologies les plus courants. Si seulement les voisins les plus immédiats du vainqueur sont considérées, la distance, également appelé rayon r, est 1. Si deux niveaux de voisins adjacents sont considérés, alors le rayon est égal à 2. Par exemple, dans le cas linéaire, un rayon de 1 comporte un voisin à droite et un à gauche du gagnant. Un rayon de 2 inclurait deux voisins neurones chacun à gauche et à droite du gagnant, soit un total de quatre dans le voisinage. Dans le cas d'une carte carrée, un rayon de 1 comprend tous les neurones séparés par une étape du gagnant et comprend huit neurones comme représenté sur la figure. Un voisinage hexagonal est associé à une carte où les neurones sont disposés dans une grille hexagonale. Pour un rayon de 1, cela comprend six neurones, un rayon de 2 comprendra une autre couche de neurones situé une étape supplémentaire de distance[12].

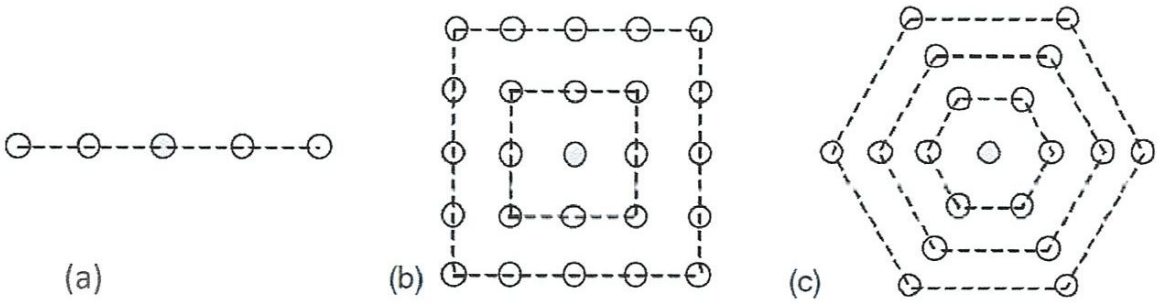


Figure II.7 : les différentes topologies de voisinage, (a) linière, (b) rectangulaire et (c) hexagonal

II. 5 Utilisation de la SOM :

Les cartes de Kohonen peuvent être utilisées dans le cadre de la projection de données multivariées, d'approximation de densité ou de classification. Elles ont été utilisées avec succès en reconnaissance de la parole, classification et traitement d'images, robotique, contrôle de processus, l'aide à la décision et l'optimisation. Les paramètres à définir dans ce cas sons :

- La topologie de la carte.
- Le nombre de cellules dans chaque dimension.
- La forme du voisinage et la fonction de réduction de la taille du voisinage.
- Le pas d'apprentissage et la fonction de réduction du pas d'apprentissage.

II. 6 Avantages et inconvénients de la SOM :

La SOM profite des relations de voisinage dans la grille pour réaliser une discrétisation dans un temps très court. On suppose que l'espace n'est pas constitué de zones isolées, mais de sous-ensembles compacts. Donc en déplaçant un vecteur référent vers une zone, on peut se dire qu'il y a probablement d'autres zones dans la même direction qui doivent être représentées par des vecteurs référents. Cela justifie le fait de déplacer les neurones proches du vainqueur dans la grille dans cette même direction, avec une amplitude de déplacement moins importante. L'algorithme présent des opérations simples, il est donc très léger en termes de coût et de calculs.

Le voisinage dans les cartes auto organisatrices est malheureusement fixe, et une liaison entre neurones ne peut être cassée même pour mieux représenter des données discontinues. Les Growing Cell Structure, ou Growing Neural Gas sont la solution à ce problème. Des neurones et les liaisons entre neurones peuvent être supprimées ou ajoutées quand le besoin s'en fait sentir [12].

II. 7 Conclusion :

Contrairement aux méthodes classiques qui ont montré leurs limites, les réseaux de neurones ont montré leurs tendances à s'adapter à des problèmes complexes grâce à leur grande capacité de calcul et d'apprentissage. Ils sont l'objet d'utilisation dans les différents domaines tels que : La reconnaissance des formes et le traitement des images.

Le grand avantage des réseaux de neurones de Kohonen est que ces derniers sont léger en cout et en calcule et son portable dans différents domaines, ce qui les rend les plus simples à utiliser et les plus rapides.

Chapitre III

Application

III.1 Introduction

Dans ce chapitre nous appliquons la SOM en premier temps sur un exemple synthétique bidimensionnel et nous évaluons les performances de la SOM en fonction du pas d'apprentissage et en fonction de la fonction de voisinage. Nous appliquons ensuite la SOM sur la base de données : tissus du sein qui est basée sur l'analyse de la variabilité des impédances observée dans le tissu normale et pathologique du sein.

III.2 Application sur un exemple synthétique :

Pour évaluer les performances de la SOM, nous considérons l'exemple synthétique de la figure III.1. C'est un Problème de classification bidimensionnel. Nous effectuons un test avec une carte 1D et un autre test avec une carte 2D

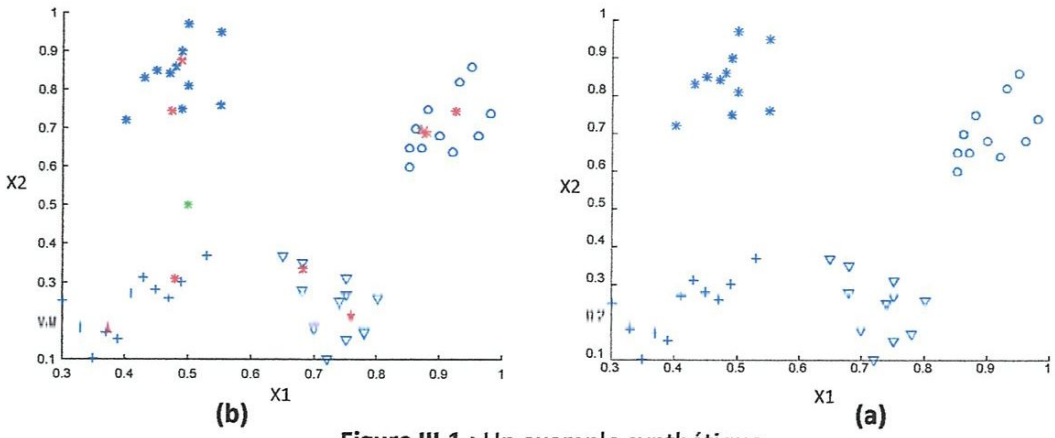


Figure III.1 : Un exemple synthétique

(a) Avant apprentissage et **(b)** Après apprentissage

III.2.1 Test avec une carte 1D :

Pour évaluer l'effet du pas d'apprentissage sur la SOM nous réalisons plusieurs tests en utilisant de différentes valeurs : 0.1, 0.2, 0.5, 0.9. Nous avons effectué des tests sur une SOM avec 8 neurones (figure III.2). La figure III.3 illustre l'évolution des erreurs au cours de l'apprentissage pour ces différentes valeurs. Nous remarquons que lorsque on réduit la valeur du pas d'apprentissage l'erreur tant vers une valeur minimale.

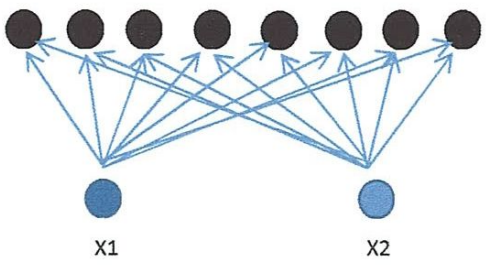


Figure III.2 : SOM 1D avec 8 neurones

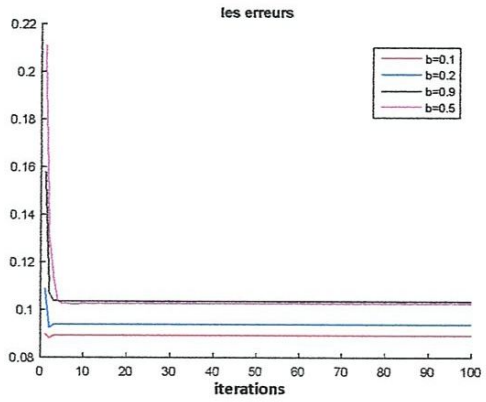


Figure III.3 : L'évolution des erreurs au cours de l'apprentissage pour différentes valeurs du pas d'apprentissage (b)

Pour évaluer l'effet de la fonction de voisinage sur la SOM nous réalisons plusieurs tests en utilisant de différentes valeurs de la constante de la fonction de voisinage : 0.5, 1, 2, 5. La figure III.4 représente l'évolution des erreurs au cours de l'apprentissage pour ces différentes valeurs. Nous pouvons noter qu'à chaque fois quand on augmente la valeur de cette constante, l'erreur est plus petite. A partir de 2 les erreurs sont acceptables.

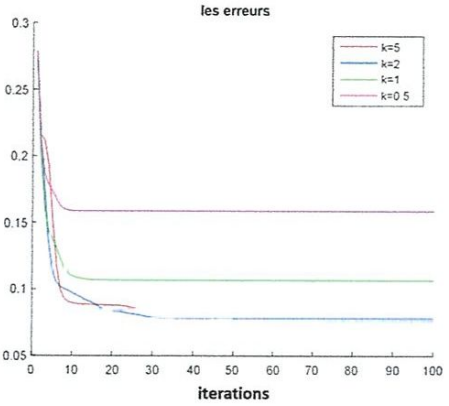


Figure III.4 : L'évolution des erreurs au cours de l'apprentissage pour différentes valeurs de la constante de la fonction de voisinage

III.2.2 Test avec une carte 2D :

Dans cette section nous appliquons une carte 2D sur l'exemple précédent. Nous utilisons des cartes avec les deux types de topologie : rectangulaire et hexagonal.

Nous utilisons dans un premier temps une carte 2D avec une topologie rectangulaire et avec un nombre de neurones : 4*3. Nous utilisons deux phases d'apprentissage : compétition et adaptation. Les paramètres d'apprentissage sont comme suit :

Nombre d'itération de la phase de compétition 100
Nombre d'itération de la phase d'adaptation 100
Pas d'apprentissage de la phase de compétition 0.2
Pas d'apprentissage de la phase d'adaptation 0.01

La figure III.5 donne une visualisation des entrées (plan des entrées). Chaque plan permet de représenter la distribution du vecteur poids d'une entrée vers tous les neurones de la grille. Dans cette représentation les couleurs représentent les valeurs des poids allant du noir au rouge ; le noir indique des valeurs nulles et le rouge les plus grandes valeurs.

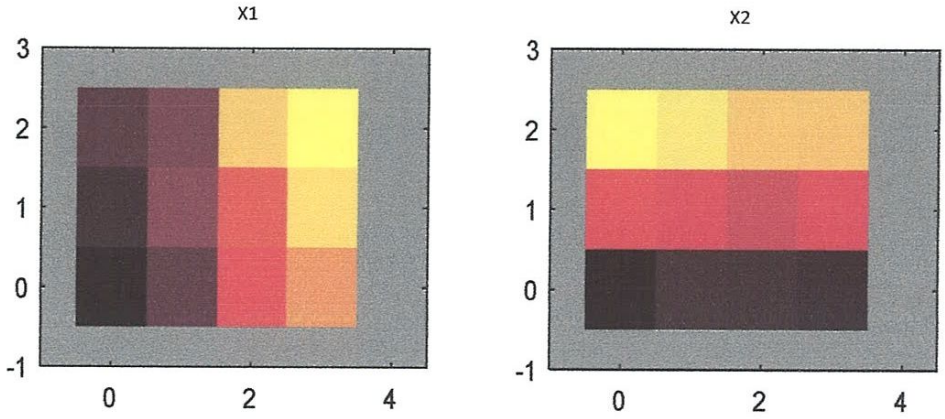


Figure III.5 : Représentation des poids d'entrées

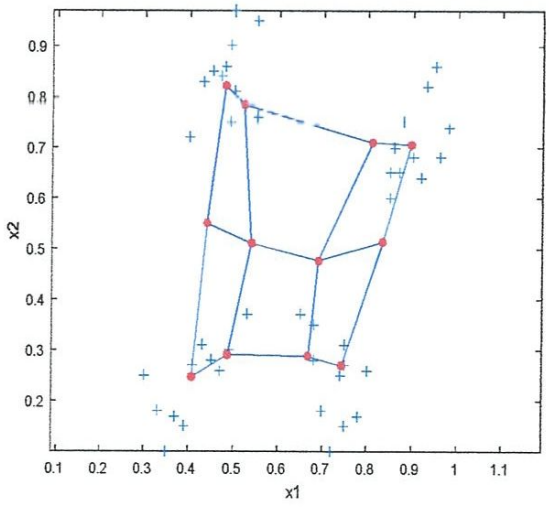


Figure III.6 : Représentation de position des poids de neurones dans l'espace caractéristique

La figure III.6 illustre la position des poids de neurones dans l'espace caractéristique avec les exemples d'apprentissage.

La figure III.7 donne le nombre des exemples d'apprentissage représentés par chacun des neurones de la grille. La taille de la partie de chaque neurone est proportionnelle au nombre d'exemples qu'il représente.

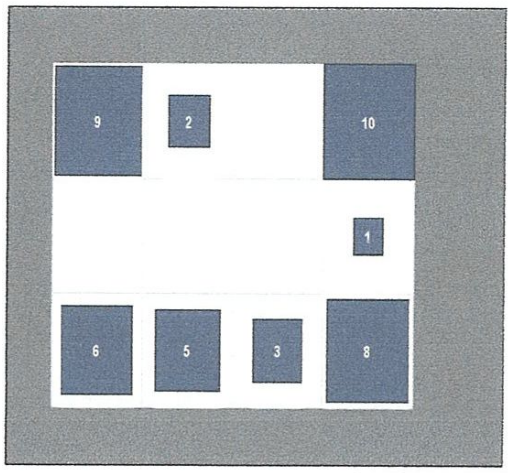


Figure III.7 : Nombre d'exemples représentés par chacun des neurones

On utilise les mêmes paramètres d'apprentissage de l'exemple précédent mais avec une topologie hexagonale.

La figure (III.8) donne une visualisation des entrées (plan des entrées). Chaque plan permet de représenter la distribution du vecteur poids d'une entrée vers tous les neurones de la grille.

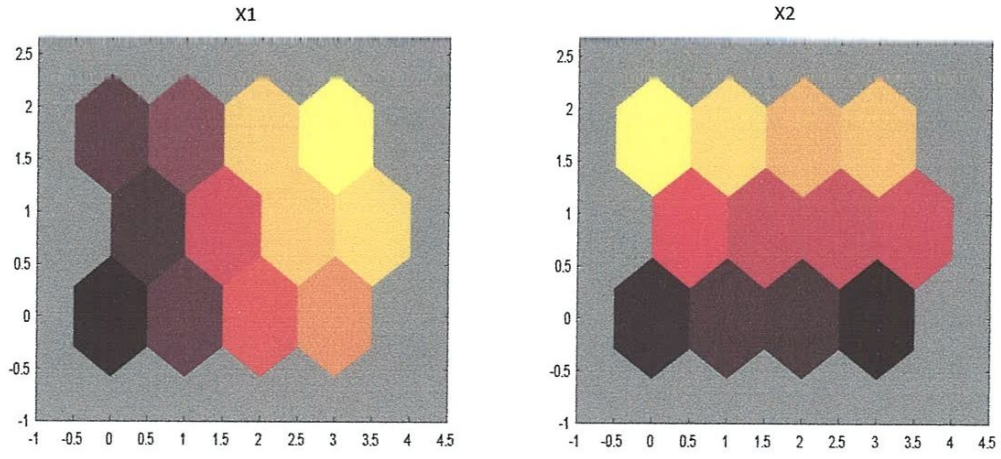


Figure III.8 : Représentation de poids d'entrée

La figure III.9 illustre la position des poids de neurones dans l'espace caractéristique avec les exemples d'apprentissage.

La figure III.10 donne le nombre des exemples d'apprentissage représentés par chacun des neurones de la grille.

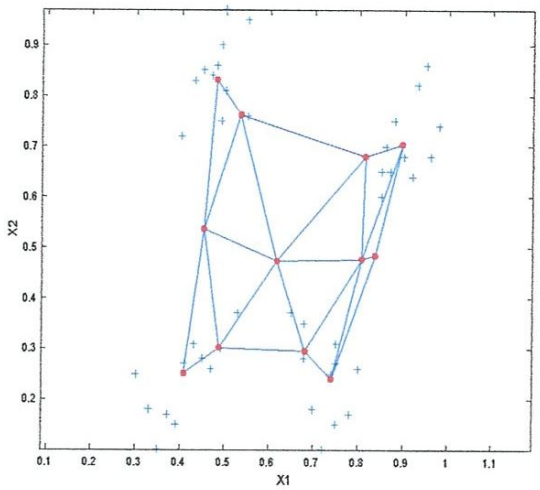


Figure III.9 : Représentation de position des poids de neurones dans l'espace

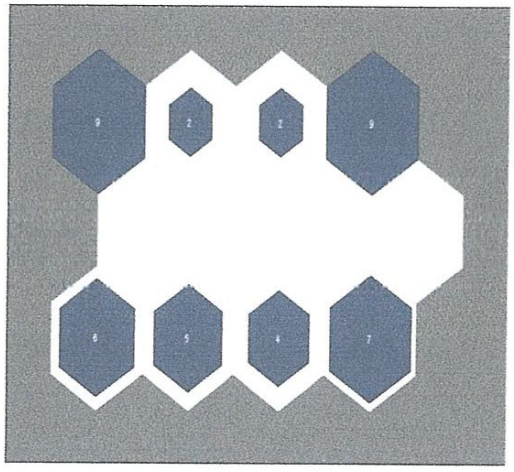


Figure III.10 : Nombre d'exemples représentés par chacun des neurones

III.3 Base de données : cancer du sein

III.3.1 Définition :

Cette base de données consiste en un ensemble de mesures des impédances électriques effectué sur des échantillons de tissus extraits des seins humain. En effet, au cours des dernières décennies des mesures électriques et diélectriques ont été effectuées dans les tissus du sein selon de différentes conditions expérimentales. Dans la gamme 488Hz-1 MHz, des différences significatives dans de l'impédance entre six groupes de tissus du sein ont été constatés [5], [6] et [7]. Ces résultats suggèrent que la spectroscopie d'impédance électrique peut être efficacement utilisée pour la discrimination des tissus du sein et en particulier pour la détection du cancer du sein.

Pour la réalisation de cette base de données, un ensemble de 106 spectres ont été enregistrés sur des échantillons de tissus provenant de 64 patients (âgés de 18 à72 ans) subissant une chirurgie mamelle. Chaque spectre est composé d'un ensemble de mesures d'impédance prises à différentes fréquences allant de 488 Hz à 1 MHz. Six groupes de tissus ont été définies avant les expériences, en fonction de la pathologie et de la morphologie du sein :

- Glande mammaire.
- Tissu conjonctif : des tissus qui séparent les cellules.
- Tissu adipeux sous-cutané adipeux (AT). Le tissu adipeux (masse grasse).
- Mastopathie : maladie bénigne et non-inflammatoire du sein (MA).
- Adénofibrome (FA). Une tumeur bénigne qu'est est la plus fréquente des tumeurs du sein.
- Carcinome (CA). (Appelé aussi épithélioma) est un cancer développé généralement sur la peau ou la muqueuse.

III.3.2 Les classes :

Tissus normaux	Glande mammaire	16 cas
	Tissu conjonctif	14 cas
	Tissu adipeux	22 cas
Tissus pathologique	Mastopathie	18 cas
	Adénofibrome	15 cas
	Carcinome	21 cas

III.3.3 Les Caractéristiques :

Les mesures d'impédance ont été faites aux fréquences en divisions successives par deux de 1 MHz à 0.488 kHz. L'impédance est l'équivalent pour le courant alternatif de la résistivité en courant continu. En raison des propriétés capacitives du tissu, le courant appliqué sur le tissu et la chute de tension créée ne sont pas en phase. L'impédance du tissu vivant est un nombre complexe exprimé soit par son module et sa phase, ou par sa partie réelle et sa partie imaginaire. Donc, ces mesures ont été tracées dans un plan complexe (réel-imaginaire) constituant ainsi le spectre d'impédance à partir duquel les caractéristiques suivantes ont été calculées.

I0 « impedivity at zero frequency »
Impédance à la fréquence zéro (résistance à la limite des basses fréquences)

PA500 Phase angle at 500 kHz
L'angle de la phase à 500 kHz

HFS high-frequency slope of phase angle (at 250, 500 and 1000 kHz points)
Pente de l'angle de la phase à hautes fréquences (250, 500 et 1000 points kHz)

DA Impedance distance between spectral ends
Distance entre les extrémités du spectre

AREA Area under spectrum
Surface sous le spectre

A/DA area normalized by DA
Surface normalize avec DA

MAX IPmaximum of the spectrum
Maximum du spectre

DR distance between I0 and real part of the maximum frequency point
Distance entre I0 et la partie réelle du point de la fréquence maximale

P length of the spectral curve
Langueur de la courbe du spectre

III.4 Application sur la base de données : cancer du sein

Vu la difficulté de séparation des classes qui caractérise généralement les données biomédicales, les auteurs de cette base de données ont suggéré pour la classification de ces données de procéder avec une approche hiérarchique à deux étapes :

Dans la première étape, deux groupes de classe ont été considérés : les tissus gras (adipeux et les tissus conjonctifs) et les quatre autres classes pris ensemble. Le problème devient donc à deux classes.

La deuxième étape consiste à classer le groupe restant à quatre classes. L'objectif principal de cette base de données consiste à séparer la classe Carcinome qui représente le cancer. Nous effectuons donc deux teste le premier est comme suit : la classe 1 : Glande mammaire, classe 2 : Mastopathie et Adénofibrome et la classe 3 : Carcinome. Ce problème est rendu donc à 3 classes.

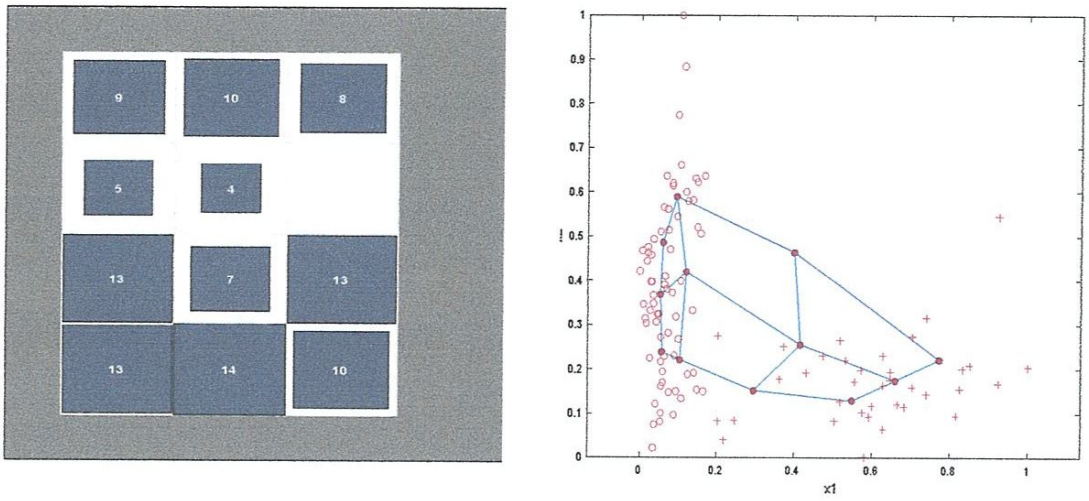
Dans un deuxième test l'objectif est de classer la classe Carcinome contre les classes restantes.

III.4.1 Résultats de la 1ère étape :

L'objectif de cette étape et de séparer les tissus gras (adipeux et les tissus conjonctifs) des quatre autres classes restant. Le problème est donc à deux classes et il est relativement simple. Les paramètres d'apprentissage sont comme suit :

Nombre d'itération de la phase de compétition 200
Nombre d'itération de la phase d'adaptation 400
Pas d'apprentissage de la phase de compétition 0.2
Pas d'apprentissage de la phase d'adaptation 0.01

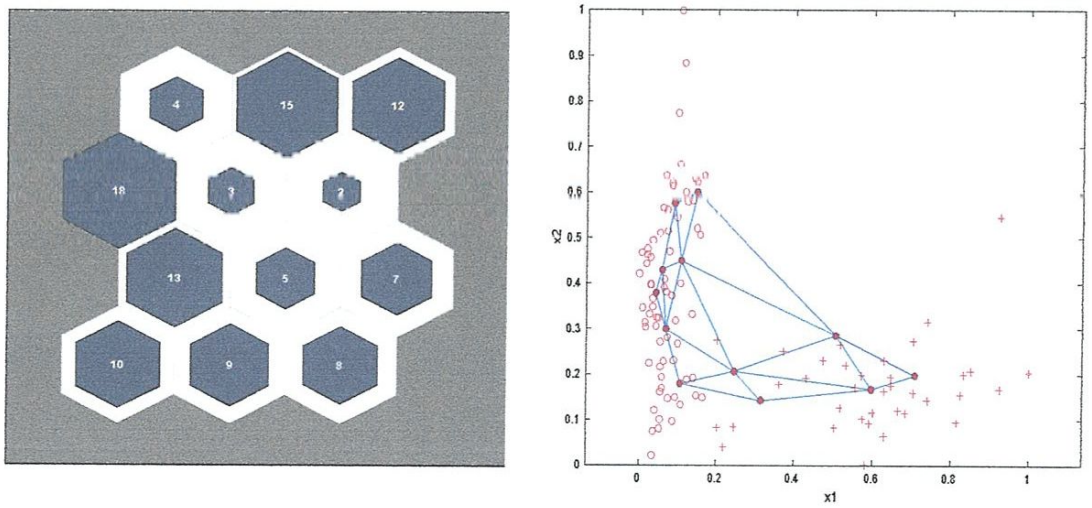
La figure III.11 représente les neurones après apprentissage de la SOM avec topologie rectangulaire : le Nombre d'exemples représentés par chacun des neurones et les positions des poids de neurones.



(a) (b)

Figure III.11 : Représentation des neurones après apprentissage de la SOM avec topologie rectangulaire :
(a) Nombre d'exemples représentés par chacun des neurones
(b) Positions des poids de neurones

De même la figure III.12 représente les neurones après apprentissage de la SOM avec topologie hexagonale.



(a) (b)

Figure III.12 : Représentation des neurones après apprentissage de la SOM avec topologie hexagonale :
(a) Nombre d'exemples représentés par chacun des neurones
(b) Positions des poids de neurones

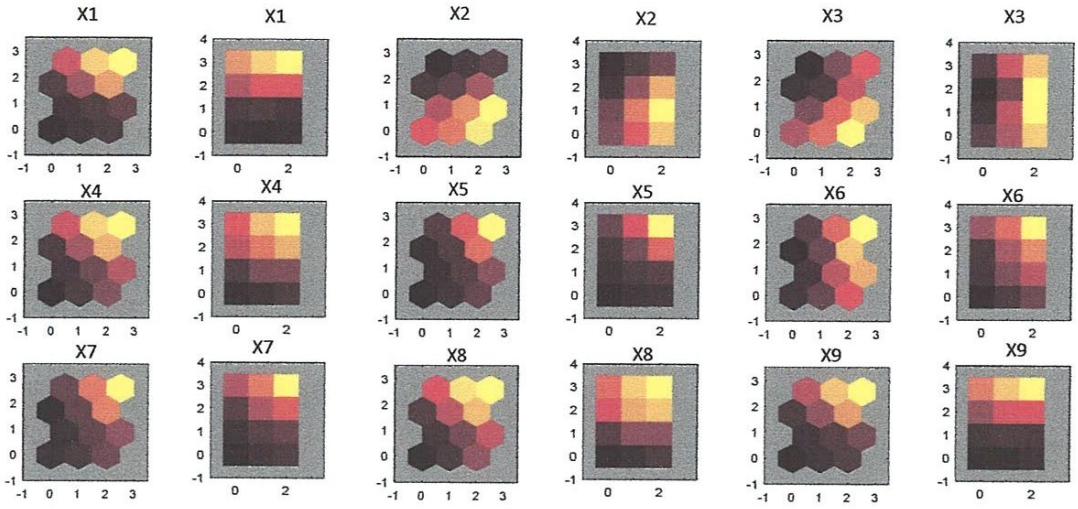


Figure III.13 : Représentation des poids d'entrées pour les deux SOM

Pour chaque caractéristique à gauche pour SOM avec topologie hexagonale et

à droite pour la SOM avec topologie rectangulaire

La figure III.13 illustre les poids d'entrées pour les deux SOM, Pour chaque caractéristique à droite pour SOM avec topologie hexagonale et à gauche pour la SOM avec topologie rectangulaire

Le tableau III.1 illustre les résultats obtenus avec les deux types de topologie : rectangulaire et hexagonal. Pour chaque type, nous effectuer plusieurs tests avec différents nombres de neurones. Nous constatons que la SOM a permis de classer les exemples de ce problème à 100% à partir de 3*4 ou 4*3 neurones.

Nombre de n / Topologie	3*3	4*3	3*4	4*4	5*5
Rectangulaire	98.1132	100	100	100	100
Hexagonale	99.0566	100	100	100	100

Tableau III.1 : Résultats de la classification de la 1ere étape obtenus avec les deux types de topologie

III.4.2 Résultats de la 2ème étape :

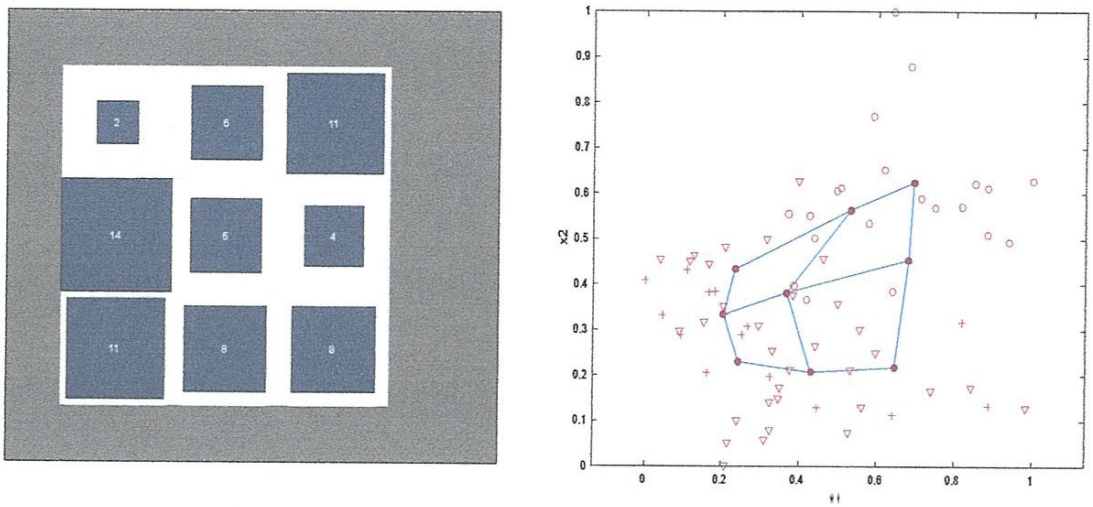
L'objectif de cette étape et de séparer le groupe restant à quatre classes selon deux façons :

a) 1er test

Dans ce test : la classe 1 : Glande mammaire, classe 2 : Mastopathie et Adénofibrome et la classe 3 : Carcinome. Ce problème est rendu donc à 3 classes. Le problème est donc à trois classes et il est plus difficile du problème précédent. Nous augmentons le nombre d'itérations comme suit :

Nombre d'itération de la phase de compétition 300
Nombre d'itération de la phase d'adaptation 600
Les pas d'apprentissage des deux phases sont les mêmes du test précèdent.

La figure III.14 représente les neurones après apprentissage de la SOM avec topologie rectangulaire : le Nombre d'exemples représentés par chacun des neurones et les positions des poids de neurones.



(a) (b)

Figure III.14 : Représentation des neurones après apprentissage de la SOM avec topologie rectangulaire :
(a) Nombre d'exemples représentés par chacun des neurones
(b) Positions des poids de neurones

Le tableau III.2 illustre les résultats obtenus avec les deux types de topologie : rectangulaire et hexagonal. Pour chaque type, nous effectuer plusieurs tests avec différents nombres de neurones.

Topologie \ Nombre de n	3*3	4*4	5*5	6*6	7*7
Rectangulaire	78.5714	80	84.2857	87.1429	90
Hexagonale	75.7143	80	81.4286	85.7143	88.5714

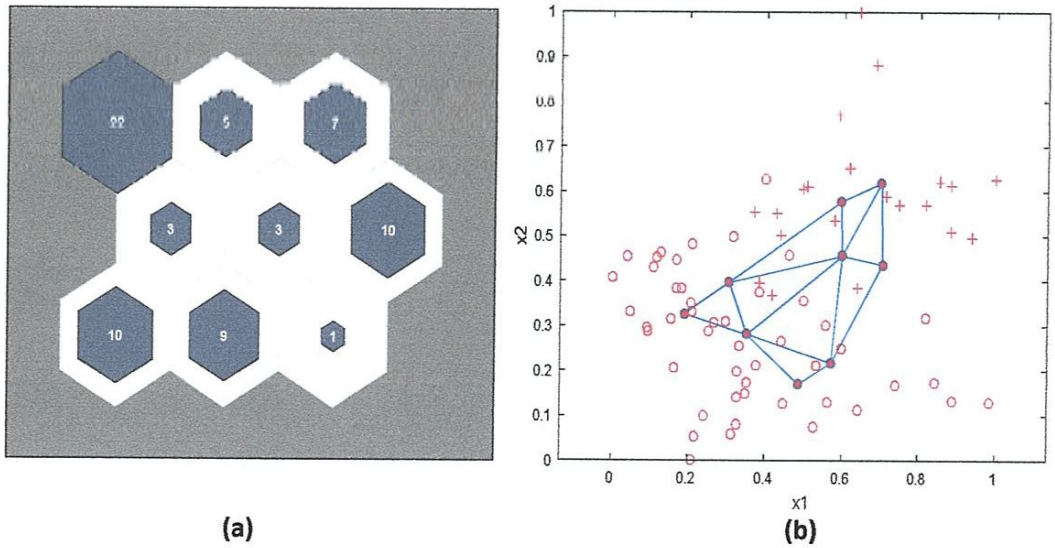
Tableau III.2 : Résultats de la classification de la 2^{éme} étape (1er test) obtenus avec les deux types de topologie

Nous constatons que la SOM n'a pas permis de bien classer les exemples de ce problème même avec un grand nombre de neurones

b) 2eme test

Dans ce test : la classe 1 : Carcinome, classe 2 : les classes restantes. Le problème est donc à 2 classes. Les paramètres d'apprentissage des deux phases sont les mêmes du test précèdent.

La figure III.15 représente les neurones après apprentissage de la SOM avec topologie hexagonale : le Nombre d'exemples représentés par chacun des neurones et les positions des poids de neurones.



(a) **(b)**

Figure III.15 : Représentation des neurones après apprentissage de la SOM avec topologie hexagonale :
(a) Nombre d'exemples représentés par chacun des neurones
(b) Positions des poids de neurones

Le tableau III.3 illustre les résultats obtenus avec les deux types de topologie : rectangulaire et hexagonal. Pour chaque type, nous effectuer plusieurs tests avec différents nombres de neurones.

Nombre de n / Topologie	3*3	4*4	5*5	6*6	7*7
Rectangulaire	92.8571	92.8571	94.2857	95.7143	97.1429
Hexagonale	91.4286	94.2857	94.2857	95.7143	98.5714

Tableau III.3: Résultats de la classification de la 2^{éme} étape (2^{éme} test) obtenus avec les deux types de topologie

III.5 Conclusion

Dans cette section nous avons effectué la classification de la base de données cancer du sein. Nous avons réalisé cette classification en deux étapes de la même façon que les auteurs de cette base de données [5], [6] et [7]. Nous avons obtenu un résultat final de 92.86 % similaire des auteurs (obtenu avec les méthodes statistique) avec une SOM de topologie hexagonale avec seulement 3*3 neurones et avec une SOM de topologie rectangulaire avec 4*4 neurones. Nous pouvons constater les bonnes capacités de la SOM.

Conclusion générale :

Dans ce travail nous avons étudié les cartes auto-organisatrices de Kohonen et leurs capacités d'analyse des données biomédicale. Nous avons appliqué ces cartes sur la base de données cancers qui se base sur l'influence des maladies sur la conduction électrique des tissus du sein.

Cette base de données comporte 6 classes représentant des tissus normaux et des tissus pathologiques. Ces maladies dont : Mastopathie, Adénofibrome, Carcinome. Les deux premières sont des maladies bénignes tandis que la troisième est un cancer développé généralement sur la peau ou la muqueuse. L'objectif principal est donc de séparer les tissus représentant cette maladie des autres tissus. Pour ce faire et vu la difficulté de séparation des classes qui caractérise généralement les données biomédicales, les auteurs de cette base de données ont suggéré la classification de ces données avec une approche hiérarchique à deux étapes : Dans la première étape, deux groupes de classe ont été considérés : les tissus gras (adipeux et les tissus conjonctifs) et les quatre autres classes pris ensemble. Dans la deuxième étape, il s'agit de séparer les tissus cancéreux (la classe V) des autres tissus.

Nous avons eu le résultat final obtenu par les auteurs (en utilisant des méthodes statistique) avec une SOM de topologie hexagonale avec seulement 3*3 neurones et avec une SOM de topologie rectangulaire avec 4*4 neurones. Ceci nous mène à constater les bonnes capacités de la SOM sur ce type donné.

Bibliographie

[1] B. Widrow and M. Hoff, Adaptive switching circuits, Convention Record, New York, 1960.

[2] Bouyedda Houcine, classification automatique supervise par les réseaux de neurones, mémoire de magister, Université de Guelma 2004.

[3] D. Hebb, The Organization of Behavior, New York: Wiley, 1949.

[4] ERIC Davalo et PATRICK Naïm, Des réseaux de neurones, EYROLLES, 1993.

[5] JE Silva, JP Marques de Sá, J Jossinet, Classification of Breast Tissue by Electrical Impedance Spectroscopy. Med & Bio Eng & Computing, 38:26-30, 2000.

[6] J. Jossinet, Variability of impedivity in normal and pathological breast tissue. Med. & Biol. Eng. & Comput, 34: 346-350, 1996.

[7] J. Jossinet, The impedivity of freshly excised human breast tissue', Physiol. Meas., 19, pp. 61-75, 1998.

[8] Juha Vesanto and Esa Alhoniemi, Clustering of the Self-Organizing Map, ieee transactions on neural networks, toronto, vol. 11, no. 3, may 2000.

[9] Haloui Adel, Reconnaissances de mots isoles arabes par hybridations de réseau de neurone, mémoire ingéniorat, Ecole Nationale d'Ingénieurs de Tunis 2005.

[10] Mohamed Nemissi, classification et reconnaissances des formes par algorithme hybrides; thèse de doctorat, Université du Guelma, 2009.

[11] Nadia Benahmed, optimisation de réseaux de neurones pour la reconnaissance de chiffres manuscrits isolés, thèse doctorat, université du Québec, 2002.

[12] R.Zaiane, Principles of Knowledge Discovery in Databases, University of Alberta,fall 1999.

[13] Taylor & Francis Group, Neural Networks for Applied Sciences and Engineering, 2006.

[14] T. Kohonen, Self-Organized Formation of topologically correct features maps, Biological Cybernetics, 1988.