UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN MATHÉMATIQUES ET INFORMATIQUE APPLIQUÉES

PAR
BENAMAR HOUMADI

# ÉTUDE EXPLORATOIRE D'OUTILS POUR LE DATA MINING

Avril 2007

# REMERCIEMENT

À la mémoire du mon père, je dédie ce modeste travail.

Mes intentions vont aux professeurs Ismail Biskri et Mhamed Mesfioui, mes directeurs de recherche pour leur aide précieuse, leur patience et l'intérêt qu'ils portaient à mon travail.

Que mes amis Zine-El-Abidine Soudani et Lamri Laoumer, acceptent mes remerciements les plus sincères pour leur collaboration et leur dévouement désintéressé.

Je remercie, aussi, les professeurs François Meunier et Lamine Mohammed Kherfi, pour avoir accepté de lire et évaluer mon mémoire.

Je ne terminerai pas sans avoir une pensée pour tout le personnel et étudiants du département de mathématiques et informatique appliquées. Je pense notamment à tous les professeurs qui font un travail extraordinaire.

À mes parents pour tout ce qu'ils m'ont donné

À ma femme Aicha pour sa patience et son dévouement

À toute ma famille

À tous mes amis.

## Sommaire

Dans le monde mouvant des technologies et sciences de l'information, de nouveaux concepts surgissent sans qu'on soit sûr de leur pérennité. Parfois ils expriment des concepts anciens qui n'ont pu se développer faute de technologies ou de maturité. Dans l'univers du décisionnel, plusieurs concepts émergent ou resurgissent grâce à l'évolution des technologies de l'information : Le *Data Mining* et l'Analyse de données [1].

Concernant le *Data Mining* qui est considéré comme un processus non élémentaire de mises à jour des relations, corrélations, dépendances, associations, modèles, structures, tendances, classes, facteurs obtenus en navigant à travers de grands ensembles de données, généralement consignés dans des bases de données (relationnelles ou pas), navigation réalisée au moyen de méthodes mathématiques, statistiques ou algorithmique[1].

On comprend, derrière le concept du *Data Mining*, l'héritage de l'intelligence artificielle et des systèmes experts. Mais on comprend aussi l'utilisation des méthodes d'analyse des données qui ont pour objet de découvrir des structures, des relations entre faits au moyen de données élémentaires et de techniques mathématiques appropriées. On ne s'étonnera pas donc de trouver au catalogue des méthodes de *Data Mining* aussi bien les réseaux de neurones, les arbres de décision.

Donc, on peut dire que la tache principale du *Data Mining* c'est utilisé des méthodes pour extraire automatiquement l'information utile de ces données et la mettre à disposition des décideurs. Toujours, et dans le même contexte, on va présenter un travail qui va nous permis de tirer l'objectif principal du *Data Mining* en basant sur la méthode des réseaux de neurones. Cette méthode se base sur l'analyse et la classification des données dans la perspective d'aider à prendre des décisions.

**CHAPITRE II : LE DATA MINING**

**CHAPITRE III : RÉSEAUX DE NEURONES**

**CHAPITRE IV : RÉSULTAT ET INTERPRÉTATIONS**

# - INTRODUCTION -

## 1. Présentation générale

En raison de l'augmentation constante du volume d'information accessible électroniquement, la conception et la mise en œuvre d'outils efficaces, permettant notamment à l'utilisateur de n'avoir accès qu'à l'information qu'il juge pertinente, devient une nécessité absolue.

Comme la plupart de ces outils sont destinés à être utilisés dans un cadre professionnel, les exigences de fiabilité et de convivialité sont très importantes. Les problèmes à résoudre pour satisfaire ces exigences sont nombreux et difficiles. Le premier cas relève du domaine de la préparation de données et le second du domaine de l'extraction d'informations.

L'accroissement de la concurrence, l'individualisation des consommateurs et la brièveté du cycle de vie des produits oblige les entreprises à non plus simplement réagir au marché mais à l'anticiper. Elles doivent également cibler au mieux leur clientèle afin de répondre à ses attentes. La connaissance de son métier, des schémas de comportement de ses clients, de ses fournisseurs est essentielle à la survie de l'entreprise, car elle lui permet d'anticiper sur l'avenir.

Aujourd'hui, les entreprises ont à leur disposition une masse de données importante. En effet, les faibles coûts des machines en termes de stockage et de puissance ont encouragé les sociétés à accumuler toujours plus d'informations. Cependant, alors que la quantité de données à traiter augmente énormément - l'institut EDS estime que la quantité de données collectées dans le monde double tous les 20 mois - le volume d'informations fournies aux utilisateurs n'augmente lui que très peu. Ces réservoirs de connaissance doivent être explorés afin d'en comprendre le sens et de déceler les relations entre données, des modèles expliquant leur comportement.

Dans cette optique, la constitution d'un *Data Warehouse*, regroupant, sous une forme homogène, toutes les données de l'entreprise sur une longue période, offre des perspectives nouvelles aux utilisateurs, notamment en termes d'extraction de connaissances grâce aux outils de Data Mining.

## 2. Le Data Mining

### 2.1. Introduction

Traduit littéralement par " forage des données ", le Data Mining est un processus est un processus non élémentaire de mises à jour de relations, corrélations, dépendances, associations, modèles, structures, tendances, classes, facteurs obtenus en navigant à travers de grands ensembles de données, généralement consignées dans des bases de données (relationnelles ou pas), navigation réalisée au moyen de méthodes mathématiques, statistiques ou algorithmiques [1].

D'après Le Gartner Group, 1996, ce processus peut être itératif et/ou interactif selon les objectifs à atteindre (Bien que non explicitement contenu dans la définition, on considère le *Data Mining* comme un processus (le plus automatisé possible) qui va des données élémentaires disponibles dans un *Data Warehouse* à la décision en apportant à chaque étape de ce processus une plus-value informationnelle qui peut aller jusqu'au déclenchement automatique d'actions en fonction de l'information de synthèse mise à jour. On comprend, derrière le concept du *Data Mining* l'héritage de l'intelligence artificielle et des systèmes experts. Mais on comprend aussi l'utilisation des méthodes d'analyses des données qui ont pour objet de découvrir des structures, des relations entre faits au moyen de données élémentaires et de techniques mathématiques appropriées. On ne s'étonnera donc pas de trouver au catalogue des méthodes de *Data Mining* aussi bien les réseaux de neurones, les arbres dits de décision que les méthodes de visualisation multidimensionnelle.

### 2.2. Qu'est-ce que le Data Mining ?

Plusieurs définitions ont été proposées dans [3], le *Data Mining* serait :

- " la découverte de nouvelles corrélations, tendances et modèles par le tamisage d'un grand nombre de données ";
- " un processus d'aide à la décision où les utilisateurs cherchent des modèles d'interprétation dans les données " ;
- " l'extraction d'informations originales, auparavant inconnues, potentiellement utiles à partir des données " ;

- "un processus de mise à jour de nouvelles corrélations, tendances et de modèles significatifs par un passage au crible des bases de données volumineuses, et par l'utilisation de modèles d'identification technique aussi bien statistiques que mathématiques." ;
- SAS Institute définit le *Data Mining* comme " le processus d'exploration et de modélisation des gisements de données permettant de découvrir des informations/indicateurs inconnus pour obtenir des avantages concurrentiels " [1].

Généralement, on s'accorde à définir le *Data Mining* comme la découverte de connaissances dans les bases de données (*Knowledge Discovery in Database - KDD*).

- Cette découverte englobe des outils statistiques mais, les méthodes statistiques classiques sont plus descriptives et confirmatives, tandis que les méthodes du *Data Mining* sont plus exploratoires (recherche de modèles sous-jacents inconnus) et décisionnelles.

Les outils actuels de *Data Mining* reprennent souvent des outils statistiques parfaitement connus depuis longtemps (comme l'Analyse Factorielle des Correspondances - AFC, la segmentation, etc.) et les incluent dans des démarches à valeur ajoutée décisionnelle. Ces outils sont théoriquement accessibles aux " utilisateurs métiers ", non-spécialistes de la statistique, par l'emploi de logiciels spécifiques relativement conviviaux.

- Le but est de découvrir des tendances cachées dans l'amas des données (la " mine " de données) et les modèles qui les traversent. Ces outils servent à déterminer des profils de comportement, à découvrir des règles, à évaluer des risques.

### 2.3. Quels sont les objectifs des méthodes de Data Mining ?

On peut regrouper les objectifs des méthodes de *Data Mining* en quatre grandes fonctions [1] :

- **Classifier** : on examine les caractéristiques d'un nouvel objet pour l'affecter à une classe prédéfinie. Les classes sont bien caractérisées et on possède un fichier

d'apprentissage avec des exemples préclassés. On construit alors une fonction qui permettra d'affecter à telle ou telle classe un nouvel individu.

- **Estimer** : la classification se rapporte à des événements discrets (par exemple :le patient à été ou non hospitalisé). L'estimation, elle, porte sur des variables continues (par exemple : la durée d'hospitalisation).

- **Segmenter** : il s'agit de déterminer quelles observations vont naturellement ensemble sans privilégier aucune variable. On segmente une population hétérogène en un certain nombre de sous-groupes plus homogènes (les clusters). Dans ce cas, les classes ne sont pas prédéfinies.

- **Prédire** : cette fonction est proche de la classification ou de l'estimation, mais les observations sont classées selon un comportement ou une valeur estimée futurs. Les techniques précédentes peuvent être adaptées à la prédiction au moyen d'exemples d'apprentissage où la valeur à prédire est déjà connue. Le modèle, construit sur les données d'exemples et appliqué à de nouvelles données, permet de prédire un comportement futur.

## 3. Problématique

Au premier rang des technologies actuelles de l'information, le *Data Mining* (qu'on pourrait traduire par Analyse Intelligente des Données) offre une réelle possibilité d'exploiter finement, rapidement et intelligemment les données afin de permettre aux utilisateurs de mieux orienter leurs actions.

Pour celui qui connaît depuis longtemps les outils de statistique et d'analyse de données, ce phénomène peut paraître curieux. On sait depuis longtemps procéder à des classifications automatiques, construire et exploiter des modèles performants, rechercher des corrélations entre variables,...etc. On connaît même dans bien des cas l'incertitude attachée aux prévisions réalisées, ce qui permet de relativiser ou pondérer les prises de décisions correspondantes.

En général les méthodes de classification s'exécutent en plusieurs étapes. L'étape la plus importante consiste à élaborer des règles de classification à partir de connaissances disponibles à priori ; il s'agit de la phase d'apprentissage. Cette dernière utilise un apprentissage soit déductif soit inductif. Les algorithmes d'apprentissage inductif dégagent un ensemble de règles (ou de normes) de classification à partir d'un ensemble d'exemples déjà classés. Le but de ces algorithmes est de produire des règles de classification afin de prédire la classe d'affectation d'un nouveau cas. Parmi les méthodes de classification utilisant ce type d'apprentissage, on cite les méthodes des k plus proches voisins, la méthode *Bayésienne*, la méthode d'analyse discriminante, l'approche des réseaux de neurones et la méthode d'arbre de décision. Dans les algorithmes d'apprentissage déductif, les règles d'affectation sont déterminées à priori par l'interaction avec le décideur, ou l'expert. À partir de ces règles on détermine les classes d'affectation des objets. Parmi les méthodes utilisant ce type d'apprentissage, signalons à titre d'exemple le raisonnement à base de cas et les ensembles approximatifs. De même, certains problèmes de classification nécessitent de combiner les deux types d'apprentissages (inductif et déductif). C'est le cas par exemple des problèmes de défaillances des machines ou du problème de diagnostic dans les images médicales.

D'autre part, la complexité des algorithmes utilisés reste un facteur qui pose d'énormes problèmes. Il n'est pas évident de déterminer dans des temps raisonnables par exemple l'unité d'information qui va nous permettre de traiter les différents types de données.

Dans ce contexte, on peut citer deux différents problèmes, ceux liés aux données et ceux liés aux classificateurs.

### 3.1. Problèmes liés aux données

La première démarche parmi les différentes démarches du *Data Mining* est la préparation des données afin de constituer une base de données qui l'on va l'utiliser pour nos tests. Cette démarche pose quelques difficultés vis-à-vis la taille et la qualité des données.
La définition de la taille de la base de données et le choix de son d'élaboration passe par un diagnostic de la qualité potentielle des données. En plus, une mauvaise qualité des

données (erreurs de saisie, champs nuls, valeurs aberrantes) impose généralement une phase de nettoyage des données.

Selon la taille et le mode de constitution de la base de données, les modalités de contrôle diffèrent :

- La base d'exemples est restreinte (moins de 300 enregistrements ou moins de 30 variables environ) et son alimentation est automatique : il est facile de contrôler de manière manuelle et visuelle chaque enregistrement pour déceler les anomalies.

- La base d'exemples est restreinte et, son alimentation étant manuelle, les risques de saisie existent : il faut contrôler la cohérence au moment de la saisie.
- La base d'exemples est importante et son alimentation est manuelle : il reste toujours un risque de saisie, mais le coût de la collecte d'informations et le délai de mise en œuvre deviennent tels qu'ils peuvent être supérieurs aux bénéfices escomptés.

Les données qui constituent notre base de données posent parfois des problèmes, ce qui ne nous permettent pas d'avoir une base vraiment prête pour la phase du test et la phase d'apprentissage, pour les valeurs manquantes, il faut bien gérer ces valeurs, en effet, l'absence des valeurs n'est pas compatible avec tous les outils de *Data Mining*, donc, soit en excluent les enregistrements incomplets, ou remplacer les données manquantes.

Dans notre cas, la base de données est constituée de données mathématiques synthétiques.

### 3.2. Problèmes liés aux classificateurs

Le deuxième problème que l'on peut rencontrer, c'est le choix du classificateur, et de savoir si le classificateur qui a été choisi s'adapte bien aux données qu'on a préparées.

Le rôle d'un classificateur est de déterminer parmi un ensemble fini de classes, à laquelle appartient un objet donné. Un classificateur doit être capable de modéliser aux mieux les frontières qui séparent les classes les unes des autres. Dans notre cas, nous avons opté pour un système de classification par les réseaux de neurones (modèle rétropropagation) des données mathématiques. Ce système produit, comme rendement, des classes de similitudes qui composent notre base de données.

Le choix du modèle du rétropropagation a été fait à cause de sa célébrité dans le domaine du *Data Mining*, en plus, il a vraiment prouvé son efficacité dans ce domaine depuis des années avec les différents types de données.

### 3.3. Notre contribution

Notre contribution consiste à mesurer l'efficacité de deux variantes de l'algorithme de rétropropagation, l'un supervisé et l'autre non supervisé. Pour mener à bien notre mission, nous avons isolé tous les facteurs qui peuvent influer sur les résultats obtenus pour ne garder que ceux qui nous intéressent. En ce qui concerne les problèmes des données, nous les avons évités en optant pour des données mathématiques synthétisées, ne contenant ni erreur, ni manque et qui sont d'une taille suffisante pour supporter nos expérimentations.

Pour le problème du classificateur et son adaptation aux données, nous avons délibérément opté pour un seul algorithme possédant deux variantes qui nous permettent de comparer les résultats obtenus par le modèle supervisé et le modèle non supervisé en écartant, ainsi, l'influence potentielle de l'algorithme lui-même.

## 4. Plan du mémoire

Notre mémoire est organisé en cinq chapitres. Dans le premier chapitre, nous présentons différents articles de recherches qui touchent essentiellement au *Data Mining* et la classification des données.

Nous en profitons pour donner un large aperçu sur le processus du *Data Mining*, et les différentes techniques inhérentes à celui-ci dans le chapitre deux.

Le chapitre trois est complètement consacré aux réseaux de neurones avec une explication détaillée de leur principe de fonctionnement et leurs structures, ainsi qu'une présentation du perceptron multicouche qui constitue l'architecture neuronale la plus utilisée dans le domaine de réseaux de neurones.

Le chapitre quatre porte sur les résultats obtenus après avoir appliqué la classification par l'algorithme de retropropagation et leurs interprétations.

Nous terminons notre mémoire par la conclusion du notre travail.

# CHAPITRE 1

# - ETAT DE L'ART -

## 1. Présentation générale

A l'ère de la société de l'information, la maîtrise des données dans l'entreprise est devenue un enjeu majeur dans la compétition pour développer, acquérir, conserver des parts de marché.

Toutefois, la maîtrise de l'information ne concerne pas uniquement le domaine marketing ou commercial, elle concerne également la qualité technique des produits, la qualité de service perçue par les clients, la maîtrise des processus de gestion, la qualité des logiciels et des systèmes d'informations, l'information sur les tendances technologiques, sur la concurrence et l'évolution des marchés, l'étude des besoins des clients, la gestion des ressources humaines, le management de la connaissance.....Maîtriser l'information pour être meilleur que ses concurrents ; voilà l'enjeu majeur de toute entreprise dans un contexte de concurrences mondiales.

Maîtriser l'information pour bien décider, c'est avoir les bonnes données, exploitées par les bons outils, au bon moment.

Au premier rang des technologies actuelles de l'information, le *Data Mining* (qu'on pourrait traduire par Analyse Intelligente des Données) offre une réelle possibilité d'exploiter finement, rapidement et intelligemment les données afin de permettre aux utilisateurs de mieux orienter leurs actions.

Le *Data Mining* est une technologie de valorisation de l'information et d'extraction de la connaissance, mais l'intérêt pour une entreprise est trop souvent masqué par la complexité des techniques mises en œuvre.

Le succès du concept de *Data Warehouse* et le nombre croissant de bases de données décisionnelles disponibles dans les entreprises, dynamise fortement l'offre *Data Mining*.

Cette offre tend à se démocratiser, en cherchant à rendre accessible au plus grand nombre, les divers outils du *Data Mining*. Pour cela, elle adopte de plus en plus un caractère "moderne" et "convivial", parfois "boîte noire" pour ne pas dire "boîte magique".

Pour qui connaît depuis longtemps les outils de statistique et d'analyse de données, ce phénomène peut paraître curieux. On sait depuis longtemps procéder à des classifications automatiques, construire et exploiter des modèles performants, rechercher des corrélations entre variables... On connaît même dans bien des cas l'incertitude attachée aux prévisions

réalisées, ce qui permet de relativiser ou pondérer les prises de décisions correspondantes (ce dernier point est aussi essentiel que de déterminer les décisions elles-mêmes...).

On peut cependant faire aux méthodes "traditionnelles" le reproche de ne pas avoir été vulgarisées. Le jargon qu'elles utilisent, les outils mathématiques (mal connus du grand public) sur lesquels elles s'appuient, les hypothèses préalables et validations requises pour une mise en œuvre rigoureuse... sont autant de freins à un usage répandu de ces méthodes.

Si des outils plus "récents", comme les réseaux de neurones ou les arbres de décisions, connaissent un certain succès, ils le doivent à leurs performances (dans certains domaines), mais probablement aussi à leurs qualités de convivialité, liées à une terminologie souvent plus accessible, à leur présentation résolument "pratique" et à l'occultation des mécanismes et algorithmes internes qui les régissent.

Pour autant, les problèmes de mise en œuvre, de compréhension des phénomènes et de validation des résultats subsistent. Ils sont même dans une certaine mesure amplifiés par la simplicité apparente de ces outils, qui n'incite pas toujours à la rigueur.

Une synthèse positive et optimiste des différents outils et courants pourrait consister à améliorer la convivialité des méthodes traditionnelles et à proposer un cadre méthodologique rendant plus fiable et rigoureuse l'utilisation des outils plus récents.

## 2. La recherche et les réseaux de neurones

### 2.1. Historique

L'historique de la recherche sur les réseaux de neurones est fort intéressant et il est bien rapporté dans [46].

Cette histoire s'étend sur trois cycles. Il y eut tout d'abord un premier mouvement d'intérêt dans la communauté scientifique au début des années 60 surtout suscité par les travaux de Rosenblatt sur le perceptron.

Durant cette période sombre, quelques chercheurs continuèrent néanmoins à œuvrer isolément dans ce domaine, notamment Grossberg à Boston [47], pour n'en nommer que quelques-uns.

## 2.2. Réseaux de neurones

Un intérêt majeur pour les réseaux de neurones s'est manifesté de nouveau dans la communauté scientifique vers la fin des années 80 avec la découverte d'une méthode d'apprentissage pour les réseaux de neurones. Connue sous le nom de « rétropropagation », et aussi avec le développement de réseaux à une couche dont le comportement pouvait être décrit selon le principe de la physique statistique. Depuis cette renaissance, le paradigme des réseaux neuronaux ne cesse d'attirer des chercheurs et de se développer à un rythme exponentiel. Les réseaux de neurones constituent une science jeune, en pleine expansion et multidisciplinaire qui regroupe des chercheurs provenant des sciences exactes, de psychologie, de l'informatique et du génie.

Parallèlement au développement rapide de la recherche en réseaux de neurones, la littérature sur le sujet augmenté à un rythme endiablé. Plusieurs revues avec comité de lecteur sont maintenant consacrées exclusivement à cette branche de la science. Mentionnons, entre autres, *Neural Networks, Biological Cybernetica* et *IEEE Transactions on Neural Networks*. Plusieurs revues consacrent une large place à des articles portant sur les réseaux neuroniques, entre autres : *IEEE Transactions on Pattern Analysis and Machine Intelligence, Computer Vision, Graphics and Image Processing, Proceedings of the IEEE, IEEE Signal Processing Magazine* et *IEEE Computer Magazine*.

Quelques articles sont considérés comme des synthèses particulièrement réussies. Parmi ceux-ci, on peut mentionner l'article [9], qui fut l'un des premiers articles à créer un vif enthousiasme dans la communauté scientifique vis-à-vis de la renaissance du domaine des réseaux de neurones. Une mis à jour récente de cet article synthèse par [ Hush, & Horne, 1993] qui procèdent à une description et à une évaluation de la grande majorité des modèles de réseaux neuroniques avec apprentissage supervisé. Un article récent [49] propose une synthèse des principaux modèles, incluant les plus récents, et des principales catégories d'applications. [Kohonen, 1988], [Grossberg, 1988], puis [Carpenter, 1989], ont publié la revue *Neural Networks* des articles majeurs sur les différents modèles de réseaux de neurones, en insistant sur les modèles d'inspirations biologiques. L'article [50] établit un parallèle intéressant et bien documenté entre les réseaux de neurones artificiels et l'intelligence artificielle.

Dans l'article [51], on constate qu'il établisse les bases de la théorie connexioniste, qui est fort bien illustré dans un article plus récent [52]. D'autres articles [53] & [54] & [55] & [56] & [57] peuvent aussi être d'intérêt pour s'initier au vaste domaine des réseaux de neurones.

L'article [10] présente une approche utilisée pour effectuer une classification des données textuelles, cette approche permet d'appliquer les performances classificatoires dynamique des réseaux de neurones à des corpus textuelles et produire donc des regroupements susceptibles d'interprétations sémantiques.

Pour l'article [11], il présente une classification mathématique des textes dans son application à l'analyse thématique philosophique de Descartes.

Dans un autre article [12], on trouve une méthode de classification des images basée sur une méthode radicalement nouvelle de recodage des données qui consiste à chercher des motifs fréquents dans les données qui consiste à chercher des motifs fréquents dans les données et à les utiliser pour coder les entrées.

L'article [13] la segmentation en deux classes d'images en couleurs de plants de forsythias qui a été réaliser à l'aide d'un réseau de neurones (perceptron à une couche cachée).

Concernant l'article [14], il a présenté vraiment quelques choses très importantes dans la classification. Il a montré comment modifier le critère d'apprentissage a fin de contrôler la distribution des erreurs au cours de l'apprentissage. Ce contrôle permet d'obtenir une meilleure marge dans les problèmes des classifications.

L'article [15] traite la manière d'élaborer un réseau neurones artificiels pour la classification des fonts arabes et latins au niveau du caractère ainsi que leur reconnaissance.

L'article [16] expose une classification des formes d'architectures des réseaux de neurones dans quatre grandes catégories :

- La première est en fonction du rôle des neurones (le modèle supervisé et non-supervisé)
- La deuxième est en fonction de la nature des liens entre les neurones (réseaux compétitif et non-compétitif).

On peut dire que la majorité des travaux qui ont effectue la tache de la classification ont un but commun, c'est l'interprétation des résultats obtenus pour la prise d'une décision qui complète la tache du *Data Mining*.

Puisque le réseau de neurones est considéré comme la technique la plus célèbre du *Data Mining*, on ne peut pas passer sans donner un petit aperçu sur quelques articles qui traitent ce domaine.

## 3. Data Mining

### 3.1. Définition et historique

Le "*Data Mining*" que l'on peut traduire par "fouille de données" apparaît au milieu des années 1990 aux États-Unis comme une nouvelle discipline à l'interface de la statistique et des technologies de l'information : bases de données, intelligence artificielle, apprentissage automatique« *machine learning* ».

On confondra ici le « *Data Mining* », au sens étroit qui désigne la phase d'extraction des connaissances, avec la découverte de connaissances dans les bases de données (KDD ou *Knowledge Discovery in Databases*) [43].
Comme l'écrivent ces derniers auteurs :

*« La naissance du data mining est essentiellement due à la conjonction des deux facteurs suivants :*

*l'accroissement exponentiel dans les entreprises, de données liés à leur activité (données sur la clientèle , les stocks, la fabrication, la comptabilité ...) qu'il serait dommage de jeter car elles contiennent des informations-clé sur leur fonctionnement (...) stratégiques pour la prise de décision.*

*- Les progrès très rapides des matériels et des logiciels (...)*

*L'objectif poursuivi par le Data Mining est donc celui de la valorisation des données contenues dans les systèmes d'information des entreprises. »*

Les premières applications se sont faites dans le domaine de la gestion de la relation client qui consiste à analyser le comportement de la clientèle pour mieux la fidéliser et lui proposer des produits adaptés. La recherche d'information dans les grandes bases de données médicales ou de santé (enquêtes, données hospitalières etc.) par des techniques de *Data Mining* est encore relativement peu développée, mais devrait se développer très vite à partir du moment où les outils existent.

### 3.2. Data Mining et la recherche

Depuis des années, plusieurs travaux existent et traitent ce vaste domaine « *Data Mining* », il est considéré comme le pilé principal pour la prise de la décision à partir d'un ensemble des données stocké sur les différents supports électroniques et accumulé depuis des années, dans ce contexte, on va citer quelques travaux et on va discuter sur leur contenus.

On va commencer avec le premier article [44] qui a comme but, est la conception et réalisation d'un logiciel permettant la modélisation de clients douteux, utilisant les techniques de *Data Mining* (Extraction des connaissances à partir de bases de donnes). Une telle connaissance pourrait être utilisée pour permettre aux décideurs et responsables stratégiques de prendre des décisions adéquates.

Ce travail à permet de créer un programme permettant d'aider le banquier ou le décideur de prêt, a prendre une décision en confrontons les données personnelles du demandeur a l'arbre déjà crée a partir de la base de données des clients. La méthode théorique utilisée est: la classification : qui consiste à examiner des caractéristiques d'un élément nouvellement présenté (Les informations relatives au demandeur de prêt : Age, Salaire, Situation Familiale, ...) afin de l'affecter à une classe d'un ensemble prédéfini. A partir de la base de données des clients.

L'article [45] avait pour but de montrer en quoi consiste le *Data Mining*, quels en sont les outils et quelle est la méthodologie de résolution d'un problème avec le *Data Mining*. Un processus de résolution dédié aux problèmes industriels a été plus particulièrement présenté. Les perspectives d'utilisations du *Data Mining* dans ces différents secteurs montrent qu'il est envisageable d'automatiser de nombreux processus d'extraction de l'information afin de réutiliser cette information dans des outils d'aide à la décision pouvant eux aussi être appuyés par les techniques de *Data Mining*.

Dans cet article, l'auteur a démontré que des perspectives particulièrement intéressantes du *Data Mining* en milieu industriel sont dans la génération automatique de règles de décision, le pilotage des systèmes de production et la génération automatique de pré gammes de production.

Concernant l'article [46], il présente un modèle hybride, à la fois robuste et fin, qui s'inspire des modèles neuronaux et de l'analyse linguistique informatique.

Le modèle hybride qu'il est proposé consiste en deux systèmes correspondant à autant d'étapes fondamentales : le système numérique et le système linguistique. Le premier est plus faiblement dépendant de la langue utilisée que le second.

Pour le système numérique, un filtrage numérique grossier du corpus est d'abord effectué. Il permet de classifier et de structurer le corpus en des classes de termes qui serviront d'indices de régularités d'associations le travail présente les étapes suivantes : elle commence par la préparation du lexique, suit alors une transformation matricielle du corpus puis, finalement, une extraction classificatoire par réseaux de neurones ART (*Adaptative Resonnance Theorie*).

Le système linguistique a pour rôle d'effectuer un traitement linguistique en profondeur des segments sélectionnés lors du filtrage qu'a permis le système numérique.

Un autre article [48] qui définit l'extraction de connaissances à partir de textes (ECT, "Text Mining")comme un descendant direct de l'Extraction de Connaissances à partir de Données (ECD, "Data Mining"). On remarque que ce travail propose la reconnaissance du raisonnement inductif en tant que mode normal de raisonnement. En plus, cet article analyse les problèmes posés par l'utilisation des systèmes existants : systèmes issus de la linguistique calculatoire, pour le pré-traitement des textes, et systèmes issus de la communauté de l'ECD, pour la détection inductive de relations au sein des textes.

On peut conclure d'après les différents travaux qu'on a vu précédemment sur la classification et le *Data Mining* que ces derniers, ont presque le même objectif, qui est l'extraction d'un savoir ou d'une connaissance à partir de grandes quantités de données, par des méthodes automatiques ou semi-automatiques, et l'utilisation industrielle ou opérationnelle de ce savoir.

Cette connaissance va contribuer de façon intéressante pour trouver des informations cachées dans la masse de données qui nous aident à la prise de décisions soit, pour trouver des solutions à des problèmes compliqués, ou pour transformer l'information à une connaissance...etc.

## 4. Les techniques de Data Mining

Les techniques de *Data Mining* représente une partie très importantes dans la tache de ce dernier, on va citer quelques une afin de donner une description générale sans entrer dans le détaille.

### 4.1. Analyse du panier de la ménagère

L'analyse du panier de la ménagère est un moyen de trouver les groupes d'articles qui vont ensembles lors d'une transaction. C'est une technique de découverte de connaissances non dirigée (de type analyse de clusters) qui génère des règles et supporte l'analyse des séries temporelles (si les transactions ne sont pas anonymes). Les règles générées sont simples, faciles à comprendre et assorties d'une probabilité, ce qui en fait un outil agréable et directement exploitable par l'utilisateur métier.

**Exemple :**

Le client qui achète de la peinture achète un pinceau

Le client qui achète un téléviseur achète un magnétoscope sous 5 ans.

### 4.2. Le raisonnement base sur la mémoire

Le raisonnement basé sur la mémoire (RBM) est une technique de prédiction et de classification utilisée dans le cadre de la découverte de connaissances dirigée. Elle peut être également utilisée pour l'estimation. Pour chaque nouvelle instance présentée, le système recherche le(s) voisin(s) le(s) plus proche(s) et procède ainsi à l'affectation ou estimation. L'avantage du RBM est qu'il est facile à mettre en œuvre, très stable et supporte tout type de données.

### 4.3. La détection automatique de clusters

La détection automatique de clusters est une technique de découverte de connaissances non dirigée (ou apprentissage sans supervision). Elle consiste à regrouper les enregistrements en fonction de leurs similitudes. Chaque groupe représente un cluster. C'est une excellente technique pour démarrer un projet d'analyse ou de data mining. Les groupes de similitudes permettront de mieux comprendre les données et d'imaginer comment les utiliser au mieux.

### 4.4. L'analyse des liens

L'analyse des liens est une technique de description qui s'inspire et repose sur la théorie des graphes. Elle consiste à relier des entités entre elles (clients, entreprises, ...) par des liens. A chaque lien est affecté un poids, défini par l'analyse, qui quantifie la force de cette relation.

Cette technique peut être utilisée pour la prédiction ou la classification mais généralement une simple observation du graphe permet de mener à bien l'analyse.

### 4.5. Les arbres de décision

Les arbres de décision sont utilisés dans le cadre de la découverte de connaissances dirigée. Ce sont des outils très puissants principalement utilisés pour la classification, la description ou l'estimation. Le principe de fonctionnement est le suivant : pour expliquer une variable, le système recherche le critère le plus déterminant et découpe la population en sous populations possédant la même entité de ce critère. Chaque sous population est ensuite analysée comme la population initiale. Le modèle rendu est facile à comprendre et les règles trouvées sont très explicites. Ce système est donc très apprécié.

### 4.6. Les réseaux de neurones

Les réseaux de neurones représentent la technique de Data Mining la plus utilisée. Pour certains utilisateurs, elle en est même synonyme. C'est une transposition simplifiée des neurones du cerveau humain. Dans leur variante la plus courante, les réseaux de neurones apprennent sur une population d'origine puis sont capables d'exprimer des résultats sur des données inconnues. Ils sont utilisés dans la prédiction et la classification dans le cadre de découverte de connaissances dirigée. Certaines variantes permettent l'exploration des séries temporelles et des analyses non dirigées (réseaux de Kohonen).

Cependant, on leur reproche souvent d'être une "boîte noire" : il est difficile de savoir comment les résultats sont produits, ce qui rend les explications délicates, même si les résultats sont bons.

### 4.7. Les algorithmes génétiques

Les algorithmes génétiques sont utilisés dans la découverte de connaissances dirigée. Ils permettent de résoudre des problèmes divers, notamment d'optimisation, d'affectation ou de prédiction. Leur fonctionnement s'apparente à celui du génome humain. Le principe de fonctionnement est le suivant : les données sont converties en chaînes binaires (comme les chaînes d'ADN - acide désoxyribo nucléique). Celles-ci se combinent par sélection, croisement ou mutation et donnent ainsi une nouvelle chaîne qui est évaluée. En fonction du résultat, les chaînes les plus faibles cèdent leur place aux plus fortes. Cette technique est particulièrement intéressante pour résoudre des problèmes d'affectation ou des problèmes sur

lesquels on peut poser une fonction d'évaluation car elle peut trouver des solutions optimisées parfois inexistantes dans les données d'origine.

### 4.8. Les agents intelligents ou knowbot

Les agents intelligents ou Knowbot sont des entités logicielles autonomes dont les plus récentes versions s'intègrent tout à fait dans le processus de data mining. Certains iront jusqu'à les considérer comme des outils de data mining. Certains d'entre eux, les plus élaborés, sont capables de suivre et mémoriser les mouvements, visites et achats sur Internet et permettent d'élaborer des profils d'utilisateurs pour leur faire des offres commerciales "un à un (one to one) ". L'utilisateur peut, quant à lui, lancer des appels d'offres et mises en concurrence automatiquement gérés par ces agents.

### 4. 9. Le traitement analytique en ligne (tael)

Pour terminer ce tour d'horizon, nous évoquerons ici le TAEL (traitement analytique en ligne) car bien que ne faisant pas partie du Data Mining, il s'agit d'outils d'analyse de données souvent utiles en préalable au Data Mining. Le TAEL est une manière de présenter aux utilisateurs les données relationnelles afin de faciliter la compréhension des données et des formes importantes qu'elles recèlent. Ces outils s'appuient sur OLAP, ROLAP, et MOLAP.

# CHAPITRE 2

# - Data Mining -

# 1. Le Data Mining

## 1.1. Introduction

Le terme de *Data Mining* est souvent employé pour désigner l'ensemble des outils permettant à l'utilisateur d'accéder aux données de l'entreprise et de les analyser. Nous restreindrons ici le terme de Data Mining aux outils ayant pour objet de générer des informations riches à partir des données de l'entreprise, notamment des données historiques et de découvrir des modèles implicites dans les données. Ils peuvent permettre par exemple à un magasin de dégager des profils de client et des achats types et de prévoir ainsi les ventes futures. Il permet d'augmenter la valeur des données contenues dans le *Data Warehouse*.

Les outils d'aide à la décision, qu'ils soient relationnels ou OLAP [On Line Analytical Processing], laissent l'initiative à l'utilisateur, qui choisit les éléments qu'il veut observer ou analyser. Au contraire, dans le cas du Data Mining, le système prend l'initiative et découvre lui-même les associations entre données, sans que l'utilisateur ait à lui dire de rechercher dans telle ou telle direction ou à poser des hypothèses. Il est alors possible de prédire l'avenir, par exemple le comportement d'un client, et de détecter, dans le passé, les données inusuelles, exceptionnelles.

## 1.2. Définition 1

Le terme de *Data Mining* signifie littéralement forage de données. Comme dans tout forage, son but est de pouvoir extraire un élément : la connaissance. Ces concepts s'appuient sur le constat qu'il existe au sein de chaque entreprise des informations cachées dans le gisement de données. Ils permettent, grâce à un certain nombre de techniques spécifiques, de faire apparaître des connaissances [4].

## 1.3. Définition 2

Le *Data Mining*, ou la fouille de données est l'ensemble des méthodes et techniques destinées à l'exploration et l'analyse de grandes bases de données informatiques, de façon automatique ou semi-automatique, en vue de détecter dans ces données des règles des associations, des tendances inconnues ou cachées, des structures particulières restituant l'essentiel de l'information utile tout en réduisant la quantité de données permettant d'étayer les prises de décision [4].

En bref, le *Data Mining* est l'art d'extraire des informations (ou même des connaissances) à partir des données.

Le *Data Mining* soit descriptif, soit prédictif.

- Les techniques descriptives (ou exploratoires) visent à mettre en évidence des informations présentes mais cachées par le volume de données.
- Les techniques prédictives (ou explicatives) visent à extrapoler de nouvelles informations à partir des informations présentes (c'est le cas de scoring)

## 2. Le processus de Data Mining

Plus qu'une théorie normalisée, le *Data Mining* est un processus d'extraction de connaissances métiers comportant les phases principales suivantes :

### 2.1. Phase 1 : Poser le problème

Cette première phase est celle où l'on expose le problème et où l'on définit les objectifs, les résultats attendus ainsi que les moyens de mesurer le succès de l'étape de data mining. Il s'agit de comprendre le contexte de la recherche en vue de donner une signification logique aux variables.

La pose du problème se décompose en deux étapes : la formulation du problème et la typologie du problème.

#### 2.1.1. La formulation du problème

La première étape consiste à formuler le problème réel sous une forme qui peut être traitée par les techniques et les outils de modélisation.
Une des approches les plus communes consiste à découper le problème complexe en sous - problèmes de complexité moindre et à collecter les données nécessaires au traitement de chacun des sous - problèmes.

#### 2.1.2. La typologie du problème

■ **Affectation :** Lorsque l'on connaît l'appartenance des éléments a une ou plusieurs classes, il s'agit d'identifier des facteurs d'affectation.
■ **Structuration :** Si l'objectif est de mettre en évidence des classes ou des facteurs de différenciation, la démarche relève alors d'une action d'identification des facteurs de structuration.

■ **Les résultats attendus**

Avant de se lancer dans un processus de *Data Mining*, il faut savoir ce que l'on attend et ce que l'on compte faire de la connaissance.

Le lancement d'un projet de data mining doit s'accompagner d'une démarche d'analyse critique des processus liés a l'exploitation des résultats.

### 2.2 Phase 2 : La recherche des données

Il s'agit dans cette phase de déterminer la structure générale des données ainsi que les règles utilisées pour les constituer.

### 2.3. Phase 3 : La sélection des données pertinentes

Le meilleur moyen de créer un modèle est de rechercher des événements similaires dans le passé. Il faut donc constituer, à partir de la mémoire de l'entreprise, cette base d'informations qui va permettre de construire l'apprentissage.

#### 2.3.1. Échantillon ou exhaustivité

L'analyste doit choisir entre étudier l'exhaustivité de la base de données et travailler sur un échantillon. Ce choix dépend en partie des outils utilisés, de la puissance des machines disponibles, du budget alloué et du niveau de fiabilité recherché.

#### 2.3.2. Le mode de création de l'échantillon

Il faut déterminer si l'échantillon doit être représentatif de la population (avec un tirage aléatoire) ou s'il doit permettre de stratifier la population en fonction de certaines sous populations. La taille des échantillons doit être déterminée en vue d'assurer la représentativité des résultats, vérifiable par des tests statistiques.

Des analyses sur une base exhaustive présentent, bien sûr, une meilleure qualité de résultats. D'une manière générale, l'exhaustivité est réservée à certains « gros détenteurs de données », tandis que le recours aux échantillons convient pour la majorité des opérations et présente des avantages certains en termes de maniabilité et de temps de réponse.

### 2.4. Phase 4 : Le nettoyage des données

La définition de la taille de la base d'exemples et le choix de son d'élaboration passent par un diagnostic de la qualité potentielle des données. Une mauvaise qualité des données (erreurs de saisie, champs nuls, valeurs aberrantes) impose généralement une phase de nettoyage des données.

### 2.4.1. L'origine de données

Selon la taille et le mode de constitution de la base de données, les modalités de contrôle diffèrent :

- La base d'exemples est restreinte (moins de 300 enregistrements ou moins de 30 variables environ) et son alimentation est automatique : il est facile de contrôler de manière manuelle et visuelle chaque enregistrement pour déceler les anomalies.

- La base d'exemples est restreinte et, son alimentation étant manuelle, les risques de saisie existent : il faut contrôler la cohérence au moment de la saisie.
- La base d'exemples est importante et son alimentation est manuelle : il reste toujours un risque de saisie, mais le coût de collecte d'informations et le délai de mise en œuvre deviennent tels qu'ils peuvent être supérieur aux bénéfices escomptés.

### 2.4.2. Les valeurs manquantes

La deuxième étape vise à gérer les données manquantes. En effet, l'absence des valeurs n'est pas compatible avec tous les outils de Data Mining.

Il faut gérer ces valeurs manquantes selon l'une des méthodes suivantes :

- **Exclure les enregistrements incomplets :** Cette première méthode, très restrictive, consiste à exclure tous les enregistrements dont une valeur manque.

- **Remplacer les données manquantes :** La deuxième méthode, supportée par certains logiciels, remplace la donnée absente par une valeur qui est soit choisie par l'utilisateur (remplacée par la moyenne ou la médiane, par exemple), soit calculée (remplacée par le résultat d'une formule de score), soit héritée.

- **Gérer les valeurs manquantes :** Lorsque l'absence de données est acceptable du point de vue de la performance du modèle, les algorithmes offrent généralement la possibilité de gérer a part la valeur manquante en la distinguant des valeurs renseignées, ou celle considérer la valeur manquante comme un facteur d'indécision.

### 2.4.3. Les valeurs nulles

La troisième étapes s'intéresse aux valeurs nulles : le nettoyage des données doit intégrer une analyse spécifique des exemples à zéro.

L'analyse de l'existence de ces enregistrements totalement nuls doit être menée afin d'en identifier les causes externes, avant de lancer les algorithmes d'apprentissage.

### 2.4.4. Prévenir la non-qualité des données

La mauvaise qualité des données complexifie l'apprentissage et nuit à la performance du modèle. Pour faire face à ce problème, certains outils intègrent du bruit (variation aléatoire d'une donnée) ou des processus « flous » (variation paramétrée) à la phase d'apprentissage. Pour cela, le logiciel simule le bruit en faisant varier les données en entrée et mesure la stabilité du modèle sur des échantillons de test.

### 2.5. Phase 5 : Les actions sur les variables

Maintenant que les variables sont pertinentes et que les données sont fiables, il faut les transformer pour préparer le travail d'analyse. Il s'agit d'intervenir sur les variables pour faciliter leur exploitation par les outils de modélisation. Ces transformations peuvent être de deux types, selon qu'elles modifient une ou plusieurs variables.

### 2.5. 1. La transformation monovariable

**■ La modification de l'unité de mesure**

Afin d'éviter certaines disproportions dans les systèmes d'unités des variables, il est recommandé de procéder à une normalisation des distributions. La normalisation sert à obtenir des ordres de grandeur comparables pour chaque variable. Elle consiste à soustraire de chaque valeur la valeur moyenne de l'échantillon et à diviser cette différence par l'écart type constaté sur l'échantillon. Une autre méthode consiste à effectuer une transformation logarithmique de la variable afin de limiter l'impact de certaines valeurs exceptionnelles.

**■ La transformation des dates en durées**

Le système de production stocke généralement des dates. Or, ces dates absolues ont en principe beaucoup moins de valeur, en matière de modélisation, que des fréquences ou des écarts entre dates.

**■ La conversion des données géographiques en coordonnées**

Les techniques de Data Mining ont généralement des difficultés à appréhender les codes postaux ou les départements. Cela tient, d'une part, à la multiplicité des codes et, d'autre part, au caractère aléatoire des codifications. Une approche habile consiste à adjoindre les coordonnées de longitude et de latitude (*méthode de géocodage*)

Le géocodage est une technique de géomarketing qui transforme des adresses ou des éléments d'adresses en coordonnées géographique.

### 2.5.2. La transformation multivariable

Elle concerne la combinaison de plusieurs variables élémentaires en une nouvelle variable abrégée. Les types de transformation sont multiples.

■ **Les ratios**

La mise en relation de deux indicateurs sous forme de ratio permet de contourner la faiblesse de certains logiciels ou de certaines techniques de modélisation.

■ **La fréquence**

Le suivi des données dans le temps permet de mesurer la répétitivité des échanges : le nombre de commandes sur les x dernières périodes.

■ **Les combinaisons linéaires**

L'expression de certains concepts se construit avec les experts par la mise en place d'indicateurs combinant des données primaires. Ainsi, le domaine du crédit, le revenu minimum de survivance, c'est-à-dire la part de revenu résiduel après déduction de toutes les charges récurrentes.

■ **Les combinaisons non linéaires**

Les boursiers nous ont habitués au calcul d'indicateurs composites complexes à base de formules non linéaires.

C'est en effet dans le domaine de la prédiction de cours que l'on trouvera le plus souvent des agrégations de variables par des formules non linéaires.

### 4.6. Phase 6 : La recherche du modèle

L'étape de recherche du modèle, qu'on appellera aussi phase de modélisation, consiste à extraire la connaissance utile d'un ensemble de données bruitées et à la présenter sous forme synthétique.

### 2.6.1. L'apprentissage

La recherche du modèle se déroule dans la phase d'apprentissage, sur une base de données d'apprentissage qui doit être distincte de la base de test.

Les bases d'apprentissage et de tests sont généralement créées à partir du même fichier de données, mais elles comprennent des enregistrements différents. La base d'apprentissages sert à construire le modèle, la base de tests sert à vérifier la stabilité du modèle.

### 2.6.2. L'automatisme et l'interactivité

Les modules construits de manière totalement automatique sont particulièrement sensibles à la qualité des données qui leur sont fournies; aussi les logiciels proposent- ils souvent une interactivité entre la machine et l'utilisateur destinée à guider et à améliorer le raisonnement au fur et à mesure de constitution du modèle.

Les outils nécessitant ou autorisant une intervention humaine demandent à l'utilisateur, pour qu'il puisse comprendre et orienter la recherche, des connaissances plus approfondies des algorithmes de calcul sous-tendant l'analyse.

Cette interactivité rend le processus de recherche itératif, via un dialogue au clavier entre l'analyse et le logiciel qui conduit l'analyste à formuler de nouvelles interrogations.

Ces itérations conduisent à affiner la recherche et élaborer des nouvelles variables.

Cette interactivité entre le logiciel et l'utilisateur contribue également à bâtir des modèles parfois moins performants mais souvent plus réalistes.

**■ Les algorithmes de calcul**

Le choix des algorithmes de calcul est déterminant pour la performance du modèle.

Il faut, dans un premier temps, positionner les nouveaux outils du *Data Mining* par rapport aux statistiques.

Pour positionner les différentes techniques de modélisation, nous allons voir une typologie des problématiques autour de trois grands pôles :

- ◆ La recherche des modèles à base d'équations
- ◆ L'analyse logique
- ◆ Les techniques de projection

**■ Les modèles d'équations**

Il se décompose en deux branches :

La branche issue de statistiques, qui englobe les techniques de régression linéaire ou logistique, l'analyse discriminante;

La branche issue des techniques neuronales, avec une distinction entre les réseaux de neurones, selon la technique d'apprentissage (rétropropagation, RBF, softmax,....etc.).

**■ L'analyse logique**

Elle se décompose aussi en trois branches, qui représentent trois méthodes d'inférence.

- La méthode inductive consiste à tirer une série de conclusion d'un ensemble de faits.

Toutes les conclusions ne seront pas vraies à 100%, mais la répartition des faits au sein d'une conclusion (97 % sans défaut et 3 % avec défaut) permet de construire un diagnostic.

*Florence est parfaite,*

*Sylvie est parfaite,*

*Dorothée est parfaite,*

⇨ *Toutes les femmes sont parfaites (1 % vrai)*

Les méthodes inductives ont commencé avec les techniques statistiques.

- La méthode abductive cherche à construire un diagnostic à partir d'une liste de déductions :

*Toutes les jolies femmes sont parfaites,*

*Florence est parfaite*

⇨ *Florence est une jolie femme (ou devrait l'être)*

- La dernière méthode d'inférence, la méthode déductive, cherche à partir d'une liste de faits (les prémisses), à construire un raisonnement.

Elle est utilisée dans le développement des systèmes experts pour appliquer un raisonnement grâce à l'instanciation de règles de productions.

*Toutes les femmes parfaites sont jolies,*

*Florence est parfaites,*

⇨ *Florence est jolie*

■ **Les techniques de projection**

Elle cherche à restituer une vision d'ensemble d'un problème. Les exemples sont positionnés sur des plans plus ou moins structurés.

On distingue généralement les techniques factorielles, qui associent des axes (appelés facteurs) d'un échantillon pour construire une interprétation à priori deces points, et les analyses de typologie, qui positionnent les exemples par rapport à des notions de proximité et ne permettent des regroupements qu'a posteriori.

Les techniques de projection sont très nettement dominées par les statistiques.

Toute fois, les travaux sur **les cartes de kohonen** (réseaux **de neurones non supervisés**).

La connaissance sera plus facilement accessible par la combinaison des différentes techniques qui contribuent souvent à une augmentation significative du résultat.

### 2.7. Phase 7 : L'évaluation des résultats

L'évaluation des résultats permet d'estimer la qualité du modèle, c'est –à- dire sa capacité à déterminer correctement les valeurs qu'il est censé avoir appris à calculer sur des cas nouveaux.

Cette évaluation prend généralement une forme qualitative et une forme quantitative.

### 2.7.1. L'évaluation qualitative

La restitution de la connaissance sous forme graphique ou textuelle contribue fortement à améliorer la compréhension des résultats et facilite le partage de la connaissance.

### 2.7.2. L'évaluation quantitative

■ **La notion d'intervalle de confiance**

L'intervalle de confiance i est donné par la formule :

$$i = \pm\sqrt{\frac{p(1-p)}{n}} \qquad (1)$$

Cet intervalle mesure la confiance vis-à-vis un sondage (avec n comme effectif de l'échantillon et p comme probabilité).

La précision d'un sondage ne dépend pas du rapport entre la taille de l'échantillon et la taille de la population mère.

■ **La validation par test**

À l'issue de la construction du modèle, il est théoriquement possible d'en tester la pertinence sur la base d'apprentissages évoquée à la phase 6.

Pour valider le modèle, il est donc préférable de constituer au préalable une base de tests ne servant qu'aux tests : le modèle découvre les exemples qui y figurent.

Les données de test soumises au modèle permettent de vérifier s'il est capable de classer correctement les données qu'il n'a jamais rencontrées auparavant.

La stabilité des résultats observés sur le fichier d'apprentissages et sur le fichier test est connue sous le nom de capacité de généralisation.

### 2.8. Phase 8 : l'intégration de la connaissance

La connaissance ne sert à rien tant qu'elle n'est pas convertie en décision puis en action.

Cette phase d'intégration de la connaissance consiste à implanter le modèle ou ses résultats dans les systèmes informatiques ou dans les processus de l'entreprise.

Elle est donc essentielle, puisqu'il s'agit de la transition du domaine des études au domaine opérationnel.

Dans certain cas, l'intégration informatique n'est pas nécessaire et l'écriture d'un rapport ou d'un cahier de procédure se révèle suffisante.

La plus part du temps cependant, le modèle trouvera toute son utilité s'il est implanté dans le système d'information, soit sous la forme d'une donnée (le résultat du modèle), soit sous la forme d'un traitement (l'algorithme du modèle).

Après voir préparer les données, nous avons besoin à des outils qui nous permettent d'extraire de la connaissance des données en découvrant des modèles, des règles dans le volume d'information présent dans les entreprises.

## 3. Les outils du Data Mining

Cette partie consiste à utiliser des données, à regrouper ou relier les éléments qui se ressemblent et à séparer ceux qui différent.

Il faut tout d'abord présenter les types de données et créer des fichiers d'analyse, afin d'expliciter les notions de variables dépendantes et indépendantes.

Ensuite, il faut préciser la manière dont se construisent les notions de ressemblance et de différence, à partir des concepts de similarités, de distance, de variance, d'association et de probabilité.

### 3.1. Type de données

**Définition d'une variable**

Nous appelons *variable* toute caractéristique d'une entité (personne, organisation, objet, événement,.....etc) qui peut être exprimée par une valeur numérique (mesure) ou codée (attribut).

Les valeurs que peut prendre une variable, pour l'ensemble des individus étudiés, sont appelées *modalités* de la variable.

Les informations sur le problème à résoudre se présentent souvent sous la forme de tables.

Les lignes d'une table représentent les exemples ou les cas à traiter.

Les variables parfois appelées attributs, décrivant un cas peuvent être de plusieurs types.

Dans le tableau I.1, on trouve une description des différents types de variables.

**Types de variables**

| Types de variables | Caractéristiques |
|---|---|
| Disjonctives | Elle peuvent prendre deux états (exemple :vrai ou faux) |
| Catégoriques non ordonnées | Les différentes catégories ne contiennent pas de notion d'ordre (exemple : la couleur des yeux) |
| Catégoriques ordonnées | Les différentes catégories peuvent être classées (exemple : les tranches d'âges) |
| Continues | Elles peuvent prendre des valeurs numérique sur lesquelles des calculs, tels que la moyenne, peuvent être effectués. |

**Tableau I.1.** Description de différentes variables

Les types des variables conditionnent fortement les techniques utilisées dans un processus de Data Mining.

### 3.2. Données utilisées en Data Mining

N'importe quel ensemble de données non structurées n'est pas analysable par les méthodes de Data Mining. Les objets sur les quels on peut appliquer les méthodes de *Data Mining* sont appelées tableaux de données.

Un tableau de données est un tableau à double entrée, consignant des nombres mettant en jeu deux ensembles d'objets : les lignes du tableau correspondent aux *individus* (ou entités); les colonnes des tableaux correspondent aux variables.

### 3.3. La notion de similarité

■ **La similarité sur des variables disjonctives**

On dit que deux objets A et B, décrits par p attributs, sont similaires si un maximum d'attributs sur les p attributs sont identiques entre eux.

**Exemple :**

Si l'on effectue une comparaison entre une voiture à moteur, une diligence et une calèche sur les cinq variables suivantes : comme dans Tableau I.2

| | **Voiture** | **Diligence** | **Calèche** |
|---|---|---|---|
| **Présence de roues** | Oui | Oui | Oui |
| **Présence d'un plancher** | Oui | Oui | Oui |
| **Présence de portes** | Oui | Oui | Non |
| **Présence d'un moteur** | Oui | Non | Non |
| **Présence d'un toit** | Oui | Oui | Non |

**Tableau I.2.** Comparaison entre une voiture à moteur, une diligence et une calèche sur cinq variables

Ce tableau de I.2. Permet de constater de manière intuitive que la diligence est plus proche de la voiture que la calèche.

Il est facile de se rendre compte que la voiture et la diligence ont quatre points communs alors que la calèche et la voiture n'en ont que deux.

**Remarque :**

En statique, la notion de point comme est dénommée coïncidence. Les coïncidences permettent de construire une mesure quantitative de la similarité entre des objets.

Les différents types de coïncidences :

Il existe deux types de coïncidences : les coïncidences positives et les coïncidences négatives, selon que les deux objets présentent ou non la même caractéristique, comme il est illustré dans le tableau I.3.

| **Valeur de l'attribut pour l'objet A** | **Valeur de l'attribut pour l'objet B** | **Coïncidence** |
|---|---|---|
| Oui | Oui | Positive |
| Oui | Non | Non-conductrice |
| Non | Oui | Non-conductrice |
| Non | Non | Négative |

**Tableau I.3.** Les différents types de coïncidences

Selon la manière de prendre en compte des coïncidences négatives, on obtiendra différentes formules, et donc différentes valeurs de similarité.

L'approche la plus restrictive, celle dite de *Russel*, n'accorde aucun poids aux coïncidences négatives [1].

Elle consiste à considérer comme le seul élément comparatif fiable les coïncidences positives sur le nombre de variables de comparaison.

L'approche la plus extensive accorde le même poids aux coïncidences positives et aux coïncidences négatives, soit la somme de toutes les coïncidences sur le nombre de variable de comparaison. Cet indice, l'indice de *Sokal* [1].

Une approche intermédiaire consiste à accorder un poids moins important aux coïncidences négatives qu'aux coïncidences positives, soit en les soustrayant du numérateur (indice *de Jaccard*), soit en les pondérant des coïncidences positives (*indice de* Dice).

■ **La similarité sur des variables quelconques**

Compte tenu de l'hétérogénéité des variables, il s'agit ici de déterminer un indice composite de toutes les similarités sur différents critères :

- La similarité sur des variables disjonctives (oui/non est égale à 1 si les deux objets présentent la caractéristique, coïncidence positive).
- La similarité sur des variables qualitatives (bleu, vert, rouge) est égale à 1 si les deux objets présentent la caractéristique.
- La similarité sur des variables quantitatives (Euro, mètre, âge) mesure l'écart entre les deux objets de manière relative par rapport à l'étendre de la distribution de la variable.

### 3.4. La notion de distance

Compte tenu de l'hétérogénéité des types de variables exploitées dans une analyse de Data Mining, il est fréquent de procéder à des transformations préalables pour positionner les individus dans un espace multidimensionnel.

La notion de similarité trouve son complément (si ce n'est que la similarité, contrairement à la distance, n'est pas nécessairement symétrique) dans la notion de distance, qui mesure l'écart dans cet espace.

La distance s'écrit :

Distance (A,B)=1 – Similarité (A,B) (2)

Deux objets similaires ont donc entre eux une distance nulle, la distance maximale sépare deux objets différents.

Cette transformation de la similarité en distance permet de donner une représentation graphique.

Ils s'agit d'une première approche permettant de positionner des objets dans un espace plus les points sont proches, plus les individus ne sont pas similaires.

Ce prédicat est la base des techniques de classifications.

Celles-ci utilisent ce même principe de distance pour construire la classification des objets en groupes.

Un groupe s'obtient par l'agrégation de *n* objets proches. Par itération de proche en proche, ce processus de regroupement finit par classifier l'ensemble de la population.

### 3.5. Les techniques de classification

#### 3.5.1. La notion de distance et la classification hiérarchique

Il existe de multiples façons de calculer des distances, nous nous intéressons ici à la distance la plus commune, la distance Euclidienne.

La notion de distance faite intuitivement référence à l'éloignement entre les points.

La distance entre les points se calcule en utilisant les propriétés des Triangles rectangles et du théorème de Pythagore selon les quelles le carré de l'hypoténuse est égal à la somme des carrées des deux autres cotés.

La distance entre A et B, notée :

$$d(A, B)^2 = d(A, F)^2 + d(F, B)^2 \qquad (3)$$

Les algorithmes de classification regroupent pas à pas les points les plus proches pour former un nouveau groupe.

Une fois ce nouvel élément né, il faut ensuite déterminer la distance entre ce nouvel élément et les points restantS.

Le travail de regroupement permet de construire l'arbre de classification à partir des distances de regroupement. Ce graphique, appelé dendrogramme, est obtenu en reportant sur l'axe vertical les distances qui ont permis le regroupement.

Cette technique de classification est connue sous le nom de classification ascendante hiérarchique, car elle part des individus qu'elle regroupe de proche en proche pour s'étendre à la population totale.

#### 3.5.2. La notion de variance et les techniques de typologie

Certaines autres techniques statistiques (méthode de Howard et Harris) utilisent la notion de variance pour mesurer le degré d'homogénéité d'une population.

La variance est un indicateur qui mesure la variance d'une variable autour de sa moyenne.

Le meilleur moyen d'appréhender une variance est de la considérer comme une surface. Plus elle est importante, plus la distribution s'éloigne de la moyenne.

La variance permet de découper une population en sous-ensembles homogènes.

- L'algorithme suivant permet de construire une classification [1]

1-On découpe la population qui présente la plus forte variance en groupe

2- On crée un premier groupe avec les éléments qui ont une valeur inférieure à la moyenne.

3- On crée un second group avec les éléments ayants une valeur supérieure ou égale à la moyenne.

Les procédures de validation de ce découpage sont multiples. Elles s'appuient toutes sur la mesure d'un indicateur par rapport à des points spécifiques qui sont les trois centres de gravités de notre nuage de points :

- Le centre de gravité du nuage total
- Le centre de gravité du groupe 1
- Le centre de gravité du groupe 2

La variance totale de notre nuage de points se calcul comme le carré de la distance entre l'ensemble des points et le centre de gravité.

Elle peut se décomposer en trois éléments :

- La variance intraclasse du groupe 1 correspond aux écarts entre les points du groupe1 et le centre de gravité du groupe 1.
- La variance intraclasse du groupe 2 correspond aux écarts entre les points du groupe2 et le centre de gravité du groupe 2.
- La variance interclasse correspond aux écarts entre les centres de gravité du groupe 1 et groupe 2 et groupe n et le centre de gravité de l'ensemble des points.

Une bonne segmentation se juge sur la variance intraclasse (plus elle est faible, plus les points sont proches) et sur la variance interclasse (plus elle est forte, plus les groupes sont éloignés).

### 3.5.3. La notion d'association

Après avoir examiné les critères qui servent à construire des segmentations des individus, nous allons traiter des indicateurs qui permettent de regrouper les variables, notamment les associations.

Les associations se mesurent différemment selon que l'on s'intéresse à des variables quantitatives ou qualitatives.

**Remarque :**

On parle de coefficient de corrélation pour les variables quantitatives et d'indicateur du $x^2$ pour les variables qualitatives.

### 3.5.3.1. L'association sur des variables quantitatives

- **La corrélation**

La corrélation mesure la relation qui existe entre deux variables.

Le coefficient de corrélation détermine si deux variables évoluent dans le même sens, c'est-a -dire si à des valeurs fortes de l'une sont associées des valeurs fortes de l'autre (corrélation positives), ou bien si a des valeurs fortes de l'une sont associées des valeurs faibles de l'autres (corrélation négative), ou encore si les deux valeurs sont indépendantes (corrélation proche de zéro).

Le coefficient de corrélation se calcule de la façon suivante :

1-Détermination des écarts par rapport à la moyenne des deux variables afin d'observer les signes de variations.

2- Détermination des produits des écarts, qui prend un signe positif ou bien négatif

3- Sommation des produits des écarts, qui donne la covariation des variables.

4- Détermination des écarts au carré, qui permet d'apprécier la variation des variables.

5- Mise en rapport de la covariation des variables avec la variation totale.

Le coefficient de corrélation définit un degré de corrélation.

Il est compris entre -1 et +1. Il signifie que deux variables sont fortement corrélées de manière positive l'orsqu'il est compris entre 0.8 et 1, qu'elles sont fortement corrélées de manière négative entre –0.8 et –1 et qu'elle sont non corrélées entre – 0.2 et + 0.2 (on parle alors d'indépendance).

La relation qui existe entre deux variables peut être utile pour solutionner certains problèmes de prévision.

On utilise pour cela les techniques de régression.

- **La régression**

La régression permet d'analyser la manière dont une variable, dite Dépendante est affectée par les valeurs d'une ou plusieurs autres variables, appelées indépendantes.

La détermination d'une fonction de régression est relativement similaire au principe de détermination du coefficient de corrélation.

Lorsque plus d'une variable est utilisée comme variable explicative, on parle de régression linéaire multiple (multiple renvoie au fait que plusieurs variables sont employées dans la prédiction).

Une analyse de régression construit une droite (régression linéaire) ou une courbe (Kermel régression) a partir d'un ensemble d'observations, en déterminant les coefficients de la droite ou de la courbe qui illustrent le mieux les données.

La détermination de ces coefficients est obtenue par des équations algébriques qui décrivent la relation entre les données et la courbe.

#### 3.5.3.2. L'association sur des variables qualitatives

- **Le test du $X^2$**

Il s'agit d'une technique qui établit l'existence d'une relation entre deux variables qualitatives.

Le test du $X^2$ repose sur une comparaison de la fréquence de distribution de ces deux variables à une distribution théorique.

Il consiste à calculer la somme des écarts entre la distribution observée et la distribution théorique et à comparer ce résultat à une valeur prédéterminée en fonction de la complexité du tableau.

## 4. Conclusion

Le Data Mining est une méthodologie qui automatise la synthèse de connaissances à partir de gros volumes de données. L'essor de cette technologie est le résultat d'un accroissement dramatique de l'information numérique qui, de part son abondance, est sous-exploitée sans outil et expertise adéquats. Cette technologie repose sur une diversité de techniques (intelligence artificielle, statistiques, théorie de l'information, génie logiciel, bases de données,...) qui requièrent des compétences variées et de haut niveau [3].

# CHAPITRE 3

## - RÉSEAUX DE NEURONNES -

# 1. Les réseaux de neurones

## 1.1. Introduction

On peut dire que parmi les buts essentiels de la recherche scientifique est de développer des machines intelligentes qui peuvent exécuter toute tache pénible et encombrante. Parmi les technologies qui sont consacrées à ce type de recherche : *l'intelligence artificielle* et les *systèmes de neurones artificiels*. Ces derniers sont basés essentiellement sur le mécanisme de transmission nerveuse d'un être humain.

L'élément fonctionnel essentiel du système nerveux est la cellule nerveuse ou neurone qui a pour rôle d'élaborer l'information reçue et transmettre les résultats a d'autres neurones.

Le cerveau humain développe mieux les solutions intelligentes qu'un ordinateur, cependant ce dernier est rapide dans l'exécution des opérations.

Les différences entre l'ordinateur et le cerveau humain sont dues à l'architecture de chacun et les méthodes du traitement correspondantes. En vue de traitement de l'information, l'ordinateur utilise des programmes basés sur des algorithmes. Ces derniers opèrent avec des séquences d'instructions contrôlées par une unité centrale complexe, afin d'aboutir à un résultat en fonction des données emmagasinées dans des mémoires. Tandis que le cerveau utilise la notion de transformation, des représentations distribuées et parallèles. Ce dernier met en communication des milliards des neurones.

Les réseaux de neurones sont des structures (la plu part de temps simulées par des algorithmes exécutés sur des ordinateurs d'usage générale, parfois sur des machines ou même des circuits spécialisés) qui prennent leur inspiration (souvent de façon assez lointaine) dans le fonctionnement des systèmes nerveux.

Leur domaine d'application est essentiellement celui de résoudre les problèmes de classification, d'association, de reconnaissance de forme, d'extraction des caractéristiques et d'identification

Les origines de cette discipline sont très diversifiées :

En 1943, *Mc CULLOCH* et *PITIS* ont proposé le premier modèle d'un système de neurones artificiels, qui est encore largement utilisé pour expliquer comment le cerveau peut réaliser les fonctions logiques.

En 1949, *DONALD HEBB* décrit une règle sur l'apprentissage [2].

Après plusieurs développement dans les modèles des réseaux de neurones, *WEBBOS* a développé en 1974 un algorithme nommé algorithme de rétropropagation, ce qui a encouragé d'autres chercheurs à reprendre la recherche dans ce domine après longue période.

### 1.2. Définition

Un réseau de neurones est un ensemble de méthodes d'analyse et de traitements des données permettant de construire un modèle de comportement à partir de données qui sont des exemples de ce comportement. Un réseau de neurones est constitué d'un graphe pondéré orienté dont les nœuds symbolisent les neurones.

Ces neurones possèdent une fonction d'activation qui permet d'influencer les autres neurones du réseau. Les connexions entre les neurones, que l'on nomme liens synaptiques, propagent l'activité des neurones avec une pondération caractéristique de la connexion. On appelle poids synaptique la pondération des liens synaptiques.

Les neurones peuvent être organisés de différentes manières, c'est ce qui définit l'architecture et le modèle du réseau. L'architecture la plus courante est celle dite du perceptron multicouche [2].

### 1.3. Applications

Les réseaux de neurones sont essentiellement utilisés pour faire de la classification. Construit à partir d'exemples de chaque classe qu'il a appris, un réseau de neurones est normalement capable de déterminer à quelle classe appartient un nouvel élément qui lui est soumis.

### 1.4. Fonctionnement

- La construction de la structure du réseau (généralement empirique).
- La constitution d'une base de données de vecteurs représentant au mieux le domaine à modéliser. Celle-ci est scindée en deux parties : une partie servant à l'apprentissage du réseau (on parle de base d'apprentissage) et une autre partie aux tests de cet apprentissage (on parle de base de test).
- Le paramétrage du réseau par apprentissage. Au cours de l'apprentissage, les vecteurs de données de la base d'apprentissage sont présentés séquentiellement et plusieurs fois au réseau. Un algorithme d'apprentissage ajuste le poids du réseau afin que les vecteurs soient correctement appris. L'apprentissage se termine lorsque l'algorithme atteint un état stable.
- La phase de reconnaissance qui consiste à présenter au réseau chacun des vecteurs de la base de test. La sortie correspondante est calculée en propageant les vecteurs à travers le réseau. La réponse du réseau est lue directement sur les unités de sortie et comparée à la réponse attendue.

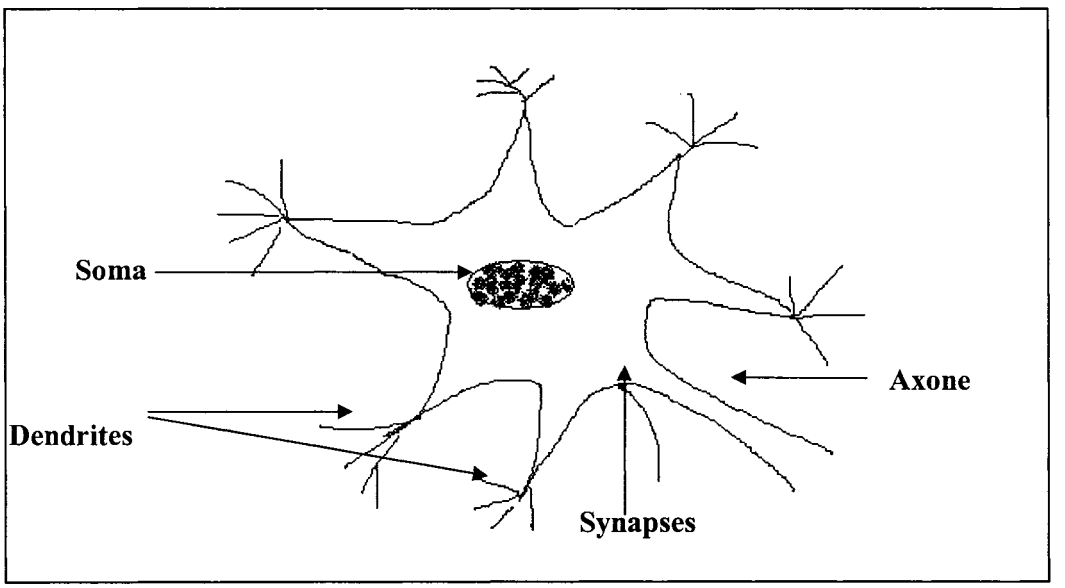
Une fois que le réseau présent des performances acceptables, il peut être utilisé pour répondre au besoin qui a été à l'origine de sa construction.

## 2. Modèle biologique

### 2.1. Définition et structure

Le bloc principal du système nerveux est le neurone. Il transmet l'information reçue vers les diverses parties du corps. Il est constitué ;

- D'un corps cellulaire nommé somma
- Des plusieurs épines semblables propagées dans le corps cellulaires nommées *dendrites*. Leur rôle est de capter les signaux qui proviennent du neurone.
- D'une seule fibre nerveuse nommé axone, qui sert à connecter le corps cellulaire aux autres neurones. L 'axone est un moyen de transport pour les signaux émis par le neurone.
- Les connexions entre les neurones se font par l'intermédiaire du corps cellulaire ou les dendrites en jonctions nommées synapses. Les synapses servent à limiter plus ou moins l'amplitude des signaux qui passent d'un neurone à un autre, comme est illustré dans la figure II.1.




**FigureII. 1.** Représentation simplifiée de neurone.

**Le modèle neuronique dont la forme la plus simple est représenté par la figure II.2 :**

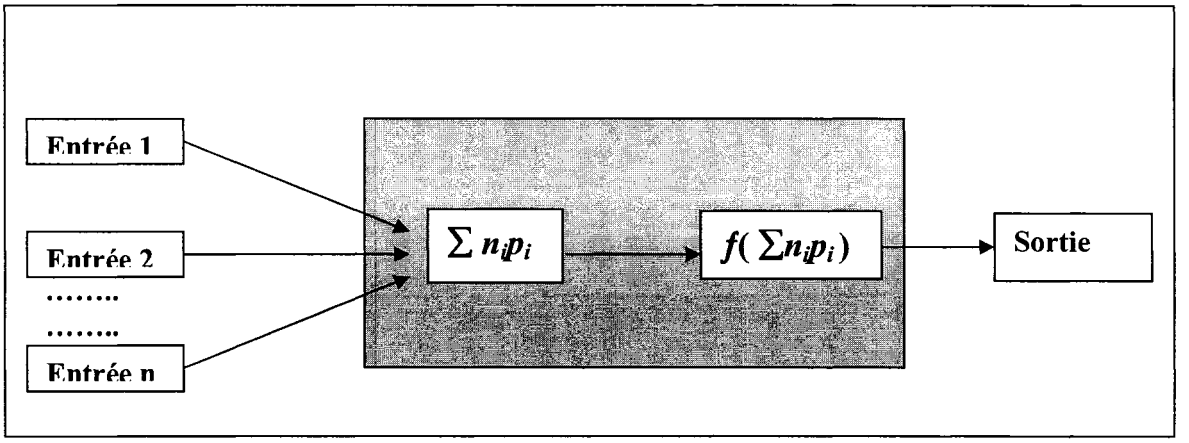



**Figure II.2.** Modèle d'un neurone artificiel.

Le figure II.2 représente l'architecture générale d'un neurone artificiel, y compris les différentes couche qui le constitue, la couche d'entrée, la couche cachée et la couche de sortie. Dans cette figure, on utilise les notations suivantes :

- $n_i$ est la valeur de sortie du nœud i du niveau précédent ( la sommation sur i correspond à l'ensemble des nœuds du niveau précédent connectés au nœud observé);
- *Pi* est le poids associé à la connexion entre le nœud i et le nœud observé;
- *f* est la fonction de transfert associée au nœud observé.

### 2.2. Fonctionnement

Le mécanisme de fonctionnement d'un neurone est de recevoir, grâce à ces dendrites, les signaux émis par les autres neurones, puis décider, à partir des données reçues, d'émettre ou non un signal à ses semblables le long de son axone.

Plus précisément, le soma recueille l'ensemble des informations reçues par les dendrites et effectue la sommation dite spatiau-temporelle. En raison de sa dimension, l'intégration somatique est aussi temporelle. Si le potentiel somatique dépasse un certain seuil, il y a émission d'un potentiel d'action ou spike. Le signal, très bref (1ms), est transmis sans atténuation le long de l'axone et réparti sur le neurone cible.

**2.3. Plasticité synaptique** (règle de HEBB)

Donald HEBB introduit la notion de *plasticité synaptique*, c'est à dire le mécanisme de modification progressive des couplages entres neurones.

D'après HEBB, le renforcement synaptique intervient lorsqu'il y a activité conjointe du neurone prè-synaptique et du neurone post-synaptique, ce qui implique chaque neurone présente deux états (*actif* ou *inactif*).

D'après cette règle, l'efficacité synaptique augmente seulement si les deux éléments sont actifs simultanément, donc elle prévoit exclusivement le renforcement des efficacités synaptiques, c'est à dire que le poids de la synapse ne peut qu'augmenter, chose qui conduit à une fatale saturation du réseau. Nous sommes donc, obligés de préciser un certain intervalle de coïncidence [2].

## 3 Étude et synthèse d'un réseau de neurone formel

La plus satisfaisante définition d'un réseau de neurone formel, est de celle de *HIECHT NILSON* : « un réseau de neurone est une structure de traitement parallèle et distribué d'informations comportant plusieurs éléments de traitement *Neurone*, qui peuvent posséder des mémoires locales et exécuter les opérations de traitements sur des informations locales. Ils sont interconnectés les uns aux autres avec des canaux des signaux unidirectionnels. »

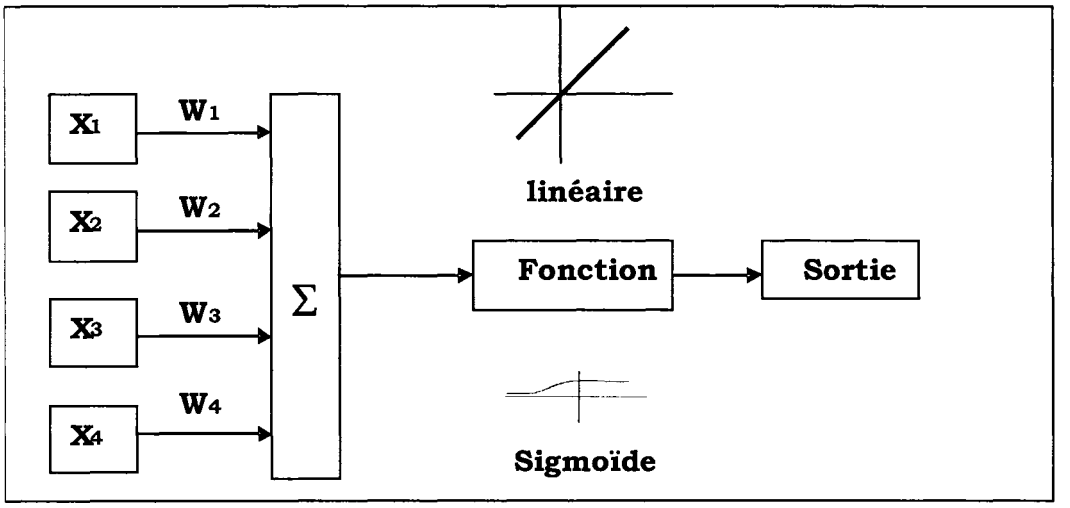
La synthèse d'un réseau de neurone formel est basée sur des caractéristiques similaires à celle d'un réseau de neurone biologique [3]. Ces caractères sont :

- Il est composé d'un nombre très grand d'éléments de traitement simple.
- Chaque élément de traitement est connecté à plusieurs éléments voisins.
- Le fonctionnement d'un réseau est basé sur le mécanisme de modification de poids de connexion pendant la phase d'apprentissage.

**a)- Neurone formel**

Un neurone formel est un petit automate qui réalise la somme pondérée des poids $W_1$, $W_2$, ……,$W_n$ des entrées $X_1$, $X_2$,……., $X_n$ qu'il reçoit du reste du réseau. Chaque nœud du réseau a un niveau d'activation numérique qui lui est associé au temps T. Ce niveau d'activation est modifié, à chaque période, par la quantité totale d'activation qu'il reçoit de ses voisins en entrée.

La figure II.3. suivant montre la structure d'un neurone formel :




**La figure II.3.** Structure du neurone formel.

**b) Fonction d'activation**

Afin de déterminer une valeur en sortie, une fonction appelée *fonction d'activation* (ou de transfert ), est appliquée à cette valeur.

La fonction d'activation la plus généralement rencontrée est une fonction sigmoïde telle que « si la somme des entrées est supérieure à un seuil, alors le neurone de sortie est activé; sinon, rien ».

La majorité des modèles utilisés aujourd'hui préfèrent employer des fonctions d'activations *continues,* qui permettent de communiquer et de traiter plus d'informations à la fois dans un seul neurone. Ceci a pour conséquence d'augmenter la puissance de calcul des réseaux.

Les étapes dans la mise en œuvre d'un réseau de neurones pour la prédiction ou la classification sont :

1- L'identification des données en entrée et en sortie ;
2- La normalisation de ces données ;
3- La constitution d'un réseau avec une structure adaptée ;
4- L'apprentissage du réseau ;
5- Le test du réseau ;
6- L'application du modèle généré par l'apprentissage ;
7- La dénormalisation des données en sortie.

### 3.1. Structure des réseaux de neurones

La structure du réseau de neurones, encore appelée « architecture » ou « topologie» du réseau de neurones, est le nombre de couches et de nœuds, la façon dont sont interconnectés les différents nœuds (choix des fonctions de combinaison et de transfert) et le mécanisme d'ajustement des poids.

#### 3.1.1. Réseau mono-couche et réseau multi-couches

On sait que l'organisation d'un réseau de neurone est constituée de couches, c'est à dire un tel réseau peut contenir une ou plusieurs couches.

**a)- Réseau mono-couches**

Dans ce type de réseau, il y a une seule couche cachée, qui relie les cellules d'association (*couche d'entrée*) aux cellules de décision (*couche de sortie*). C'est la seule couche de connexion modifiable.

Les neurones de la couche d'entrée d'un réseau mono-couche (*perceptron*) effectuent seulement un prétraitement et la classification effective est effectuée par les neurones de la couche de sortie.

Ce réseau offre une grande convergence vers la solution du problème, malheureusement sa stratégie d'apprentissage n'offre que des séparations linéaires, limitées à la seule classe de problèmes linéairement séparables.

**b)- Réseau multi-couches**

Pour surmonter les limitations d'un réseau mono-couche , on utilise un réseau multi-couche, où la sortie n'est connectée à l'entrée qu'après quelques couches de neurones intermédiaires apportant une richesse à la structure pour accroître la capacité de réseau. Notons que les couches internes n'ont aucune connexion prédéfinie, elles servent seulement à contribuer à l'obtention de résultats souhaités à la sortie.

Le problème de séparation linéaire est donc résolu. Pour obtenir une séparation linéaire, on doit tenir compte de ce qui a été dit plus haut d'une part, et le bon dimensionnement en utilisant le modèle de *rétropropagation* d'autre part.

### 3.1.2. Réseaux récurrents et réseaux non récurrents

Les réseaux de neurones sont répartis en deux grandes classes :

**a) Les réseaux non récurrents (statiques)**

Dans ce type de réseau, on utilise une structure à couche. Pour ce type de réseau les neurones de la même couches ne sont pas connectées, chaque couche reçoit des signaux de la couches précédente et transmet le résultat de ces traitement à la couche suivante. En conséquence le signal d'entrée prend un sens unique de l'entrée vers la sortie tout en ayant traversé des couches cachées.

Un réseau de neurones statique est généralement organisé en plusieurs couches de neurones appelées réseaux multicouches comme il est illustré dans la figure II.4.

**L'architecture d'un tel réseau est donnée par figure II.4**

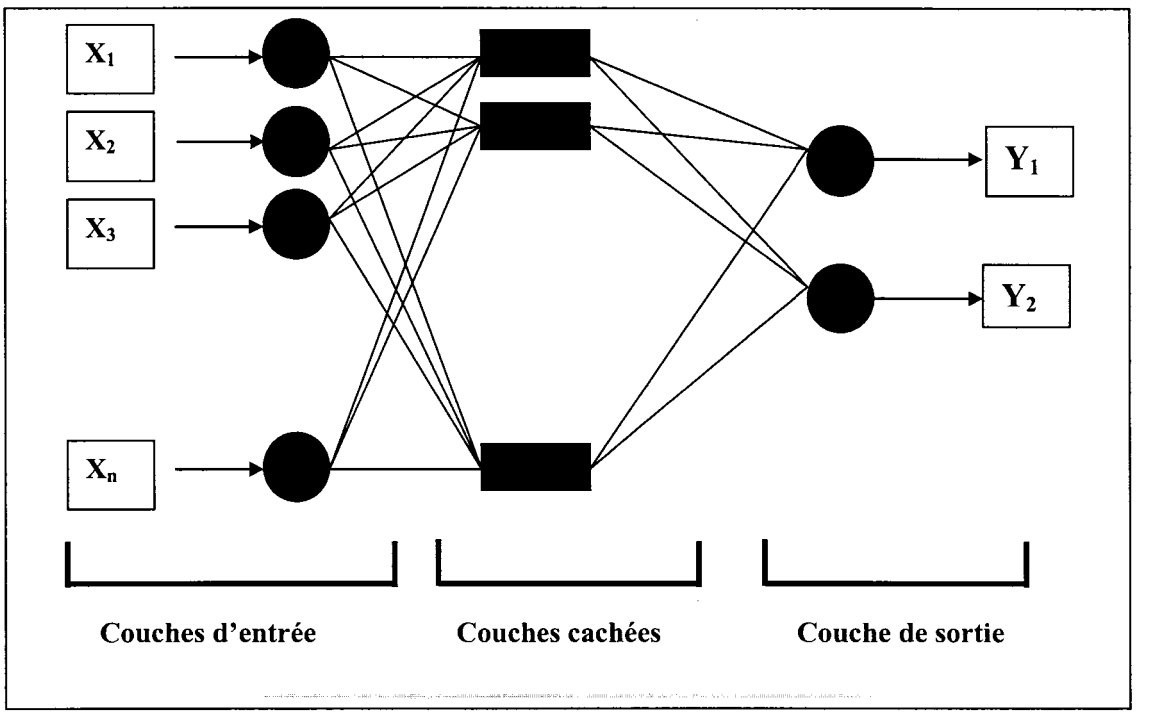



**Figure II. 4.** Architecture d'un réseau statique

Un réseau statique est constitué par :

- Une couche d'entrée qui reçoit ses signaux d'entrée du milieu externe.
- Une ou plusieurs couches cachées (*intermédiaire*)
- Une couche de sortie qui fournit les résultats de traitement du réseau.

**b) Réseaux récurrents (dynamiques)**

Dans ce type de réseaux, les neurones sont entièrement connectés et les sorties des neurones de la couche sortie sont réinjectées sur les entrées des neurones précédents d'où l'existence d'une boucle de retour, cette dernière à pour rôle d'équilibrer le système lorsqu'il est soumis à un stimulus extérieur.

Dans le réseau récurrent les décisions ne sont pas présentes instantanément, mais par étapes successives. La structure d'un réseau dynamique est donnée par la Figure II.5 synoptique de la figure suivante :

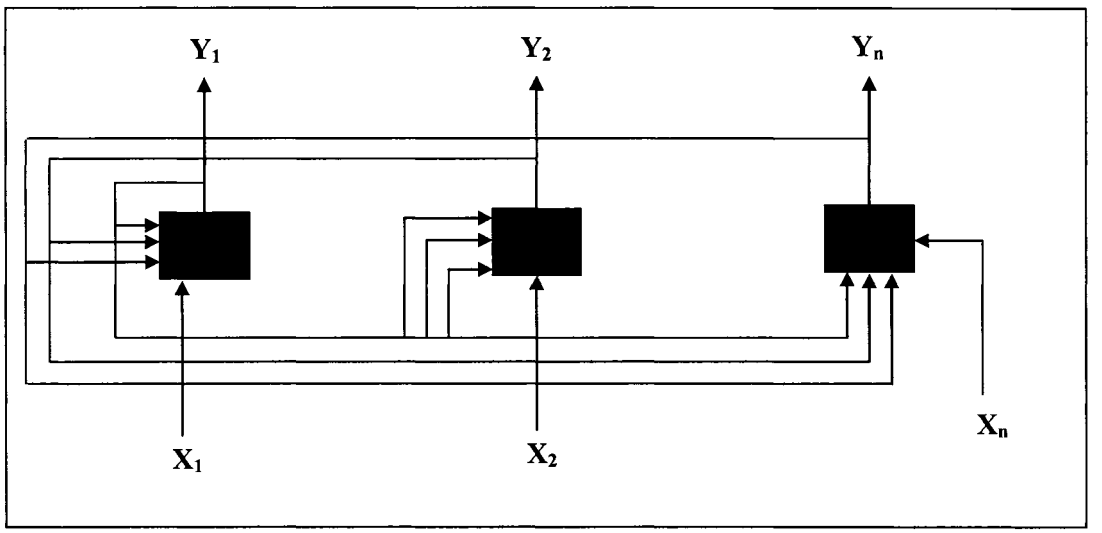



**Figure II.5.** Structure d'un réseau dynamique

### 3.1.3. Fonctionnement d'un réseau

Un réseau de neurone peut fonctionner en deux modes, parallèle ou séquentiel. Dans le mode parallèle tous les neurones calculent leurs nouvelles activations et leurs sorties, et les transmettent aux neurones auxquels ils sont connectés, à chaque top d'horloge. Contrairement, au mode séquentiel, un seul neurone calcule sa nouvelle activation et sa sortie puis les transmet aux neurones auxquels il est connecté, à chaque top d'horloge. Donc, le calcule est fait en fonction des entrées des neurones au top d'horloge précédent.

On peut obtenir d'autres modes dits *mixtes* en combinant les deux modes précédents.

### 3.1.4. Apprentissage

On peut définir l'apprentissage par la modification des interactions entre neurones, l'apprentissage consiste donc à ajouter les poids synaptiques de telle façon que le réseau présente un certain comportement désiré.

Les procédures d'apprentissage peuvent se subdiviser, en deux grandes catégories :

Apprentissage *supervisé* ou apprentissage *non supervisé.*

#### a)- Apprentissage supervisé

L'apprentissage supervisé implique l'existence d'un professeur qui a pour rôle d'évaluer le succès ou l'échec du réseau quand on lui présente un stimulus connu.

Cette supervision consiste à renvoyer au réseau une information lui permettant de faire évoluer ses connections afin de faire diminuer son taux d'échec. C'est a dire que ce professeur présente au réseau de neurones une entrée et la sortie désirée correspondante, pour faire la comparaison avec les sorties actuelles des vecteurs d'entrées. A partir de l'erreur calculée, les poids sot ajustés pour avoir des sorties correspondantes aux réponses désirées.

Ce calcul se répète jusqu'à ce que l'erreur soit minimale par rapport à un critère préalable, et par conséquent les coefficients synaptiques prennent les valeurs optimales.

#### b)- Apprentissage non supervisé

Les réseaux, utilisant l'apprentissage non supervisé, sont souvent appelés *auto-organisateurs*, ou encore à apprentissage *compétitif.* Dans ce type d'apprentissage la connaissance de la sortie désirée n'est pas nécessaire c'est à dire que le réseau s'*auto-organise* et *organise* les entrées qui sont présentées de façons à optimiser un critère de coût donné.

### 3.1.5. Choix de l'échantillon d'apprentissage

L'apprentissage du réseau de neurones sera d'autant meilleur qu'il s'effectuera sur un échantillon suffisamment riche pour représenter toutes les valeurs possibles de nœuds de toutes les couches du réseau, c'est-à-dire en particulier toutes les modalités possibles de chaque variable, en entrée ou en sortie.

Il faut aussi veiller à ce que les enregistrements analysés ne soient pas triés selon un ordre significatif.

### 3.1.6. Normalisation des données

Les données utilisées dans un réseau de neurones doivent être numériques et leurs modalités comprises dans l'intervalle [0,1], ce qui implique, quand ce n'est pas le cas, une normalisation des données. Pour que le travail de normalisation soit correct, il faut, bien entendu, que le jeu de données d'apprentissage couvre toutes les valeurs rencontrées dans la population tout entière, et, en particulier, les valeurs extrêmes des variables continues.

- **Variables continues**

Même en les normalisant, les variables continues peuvent connaître le problème d'écrasement des valeurs normales les valeurs extrêmes. Plusieurs moyens existent pour bien normaliser ce type de variable. On peut discrétiser la variable et la remplacer, par exemple, par ses quartiels. On peur normaliser, non pas la variables, mais le logarithme de cette variable, qui « distend » le début de l'échelle. On peut normaliser la variable linéairement, pour ses valeurs comprises entre –3 et +3 fois l'écart –type $\sigma$ autour de la moyenne $\mu$ , et envoyer les valeurs à $\mu - 3\sigma$ sur 0 , et les valeurs supérieures à $\mu + 3\sigma$ sur 1.

- **Variables catégoriques**

Un moyen fréquemment utilisé pour obvier à cette difficulté est d'avoir autant de nœuds que de modalités des variables catégoriques, en créant des variables binaires (appelées « indicatrices ») dont la valeur 1 ou 0 signifie que la variable catégorique a ou non cette modalité.

**Remarque :**

Avant d'utiliser un réseau de neurones sur des données catégoriques, il faut donc réduire le plus possible le nombre de modalités de ces données.

### 3.1.7. Les principaux réseaux de neurones

Les réseaux de neurones diffèrent selon :

- Les neurones utilisés
- La structure du réseau
- Le mode de calcul

Il existe différents modèles de réseaux de neurones. Les principaux, le perceptron multicouches (PMC : Multi Layer Perceptron), le réseau à fonction radiale RBF (Radial Basis Function) et le réseau de Kohenen.

Les réseaux de neurones PMC et RBF sont des réseaux à *apprentissage supervisé* ( on recherche une ou plusieurs valeurs « cible » en sortie ) : ils appartiennent à la famille des techniques *prédictives*.

Au contraire, le réseau de Kohenen est un réseau à *apprentissage non supervisé*.

Il cherche à segmenter la population en groupes distincts rassemblant des éléments similaires : il appartient à la famille de techniques descriptives [2].

# 4. Développement d'un réseau de neurones

Procédure de développement d'un réseau de neurones.

Le cycle classique de développement peut être séparé en sept étapes :

1. la collecte des données,
2. l'analyse des données,
3. la séparation des bases de données,
4. le choix d'un réseau de neurones,
5. la mise en forme des données,
6. l'apprentissage,
7. la validation.

## 4.1. Collecte des données

L'objectif de cette étape est de recueillir des données, à la fois pour développer le réseau de neurones et pour le tester. Dans le cas d'applications sur des données réelles, l'objectif est de rassembler un nombre de données suffisant pour constituer une base représentative des données susceptibles d'intervenir en phase d'utilisation du système neuronal.

La fonction réalisée résultant d'un calcul statistique, le modèle qu'il constitue n'a de validité que dans le domaine où on l'a ajusté. En d'autres termes, la présentation de données très différentes de celles qui ont été utilisées lors de l'apprentissage peut entraîner une sortie totalement imprévisible.

## 4.2. Analyse des données

Il est souvent préférable d'effectuer une analyse des données de manière à déterminer les caractéristiques discriminantes pour détecter ou différencier ces données. Ces caractéristiques constituent l'entrée du réseau de neurones. Notons que cette étude n'est pas spécifique aux

réseaux de neurones, quelque soit la méthode de détection ou de classification utilisée, il est généralement nécessaire de présenter des caractéristiques représentatives

Cette détermination des caractéristiques a des conséquences à la fois sur la taille du réseau (et donc le temps de simulation), sur les performances du système (pouvoir de séparation, taux de détection), et sur le temps de développement (temps d'apprentissage).

Une étude statistique sur les données peut permettre d'écarter celles qui sont aberrantes et redondantes.

Dans le cas d'un problème de classification, il appartient à l'expérimentateur de déterminer le nombre de classes auxquelles ses données appartiennent et de déterminer pour chaque donnée la classe à laquelle elle appartient.

### 4.3. Séparation des bases de données

Afin de développer une application à base de réseaux de neurones, il est nécessaire de disposer de deux bases de données : une base pour effectuer l'apprentissage et une autre pour tester le réseau obtenu et déterminer ses performances. Afin de contrôler la phase d'apprentissage, il est souvent préférable de posséder une troisième base de données appelée « base de validation croisée ».

Les avantages liés à l'utilisation de cette troisième base de données seront exposés dans les sections suivantes. Il n'y a pas de règle pour déterminer ce partage de manière quantitative. Il résulte souvent d'un compromis tenant compte du nombre de données dont on dispose et du temps imparti pour effectuer l'apprentissage. Chaque base doit cependant satisfaire aux contraintes de représentativité de chaque classe de données et doit généralement refléter la distribution réelle, c'est à dire la probabilité d'occurrence des diverses classes.

### 4.4. Choix d'un réseau de neurones

Il existe un grand nombre de types de réseaux de neurones, avec pour chacun des avantages et des inconvénients. Le choix d'un réseau peut dépendre :

- de la tâche à effectuer (classification, association, contrôle de processus, séparation aveugle de sources...),
- de la nature des données,
- d'éventuelles contraintes d'utilisation temps-réel (certains types de réseaux de neurones, tels que la 'machine de Boltzmann', nécessitant des tirages aléatoires et un nombre de cycles

de calculs indéfini avant stabilisation du résultat en sortie, présentent plus de contraintes que d'autres réseaux pour une utilisation temps-réel),

- des différents types de réseaux de neurones disponibles dans le logiciel de simulation que l'on compte utiliser.

Ce choix est aussi fonction de la maîtrise ou de la connaissance que l'on a de certains réseaux, ou encore du temps dont on dispose pour tester une architecture prétendue plus performante.

### 4.5. Mise en forme des données pour un réseau de neurones

De manière générale, les bases de données doivent subir un prétraitement afin d'être adaptées aux entrées et sorties du réseau de neurones. Un prétraitement courant consiste à effectuer une normalisation appropriée, qui tienne compte de l'amplitude des valeurs acceptées par le réseau.

### 4.6. Apprentissage du réseau de neurones

Tous les modèles de réseaux de neurones requièrent un apprentissage. Plusieurs types d'apprentissages peuvent être adaptés à un même type de réseau de neurones. Les critères de choix sont souvent la rapidité de convergence ou les performances de généralisation.

Le critère d'arrêt de l'apprentissage est souvent calculé à partir d'une fonction de coût, caractérisant l'écart entre les valeurs de sortie obtenues et les valeurs de références (réponses souhaitées pour chaque exemple présenté).

La technique de validation croisée, qui sera précisée par la suite, permet un arrêt adéquat de l'apprentissage pour obtenir de bonnes performances de généralisation.

Certains algorithmes d'apprentissage se chargent de la détermination des paramètres architecturaux du réseau de neurones. Si on n'utilise pas ces techniques, l'obtention des paramètres architecturaux optimaux se fera par comparaison des performances obtenues pour différentes architectures de réseaux de neurones.

Des contraintes dues à l'éventuelle réalisation matérielle du réseau peuvent être introduites lors de l'apprentissage.

### 4.7. Validation

Une fois le réseau de neurones entraîné (après apprentissage), il est nécessaire de le tester sur une base de données différente de celles utilisées pour l'apprentissage ou la validation croisée. Ce test permet à la fois d'apprécier les performances du système neuronal et de détecter le

type de données qui pose problème. Si les performances ne sont pas satisfaisantes, il faudra soit modifier l'architecture du réseau, soit modifier la base d'apprentissage (caractéristiques discriminantes ou représentativité des données de chaque classe).

## 5. Le perceptron multicouche

Les réseaux de neurones du type *perceptron multicouche* constituent sans doute l'architecture neuronale la plus utilisée dans le domaine des réseaux de neurones formels. En effet, ces réseaux ont été utilisés pour la résolution de problèmes très variés : la reconnaissance de formes, la détection de pannes, la prévision temporelle, le traitement d'images, le traitement du signal, etc. Les performances obtenues en utilisant ces réseaux constituent l'une des principales raisons de l'intérêt croissant pour les réseaux de neurones artificiels.

### 5.1. Définition

Le *perceptron* est un modèle de réseau de neurones avec algorithme d'apprentissage créé par Frank Rosenblatt en 1958.

### 5.2. Architecture

Un réseau multicouche de neurones est constitué de plusieurs couches de neurones formels adaptatifs comme il est illustré à la figure II.6. La principale difficulté pour les chercheurs résidait dans l'absence d'un algorithme pour corriger les poids des couches cachées, étant donné qu'on ne disposait pas pour ces neurones d'un signal d'erreur. Le seuil signal d'erreur qui pouvait être calculé était celui pour la couche de sortie puisqu'on connaît la valeur désirée pour chacun des neurones de sortie et la valeur effectivement affichée pour ces neurones.

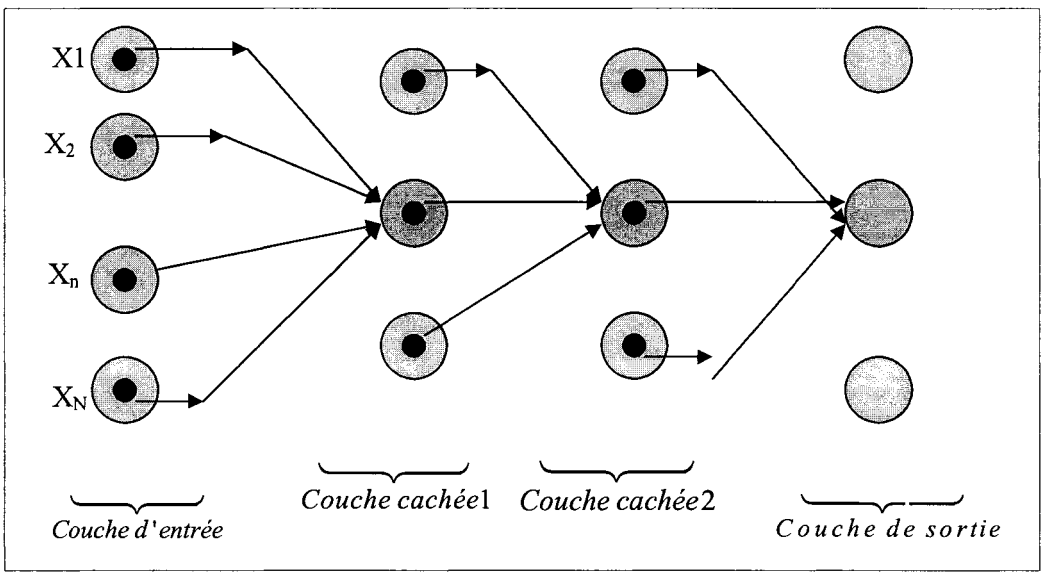



**Figure II.6** Architecture d'un réseau multicouche de neurones.

La première couche, ou *couche d'entrée*, a pour seule mission de présenter le vecteur associé à une forme à l'entrée forme à l'entrée du réseau. Par conséquent, les neurones qui la composent ne sont pas de véritables neurones du modèle de *McCulloch&Pitts* [2] et cette couche n'est généralement pas comptée. Chaque neurone ne possède qu'une seule entrée, une valeur de polarisation nulle, une fonction d'activation linéaire et une fonction de sortie également linéaire. La valeur du poids de chaque connexion est constante et unitaire.

En revanche, toutes les couches suivantes sont composées de neurones du modèle de *McCulloch&Pitts* . La dernière couche est appelée *la couche de sortie*. Toutes les couches comprises entre la couche d'entrée et celle de sortie portent le nom de *couches cachées* et sont numérotées d'une manière séquentielle.

Dans la littérature anglo-saxone, cette architecture porte le nom du *MultiLayer Perceptron* [2]. Elle correspond à une certaine réalité biologique, car la couche d'entrée peut être assimilée à la rétine, la couche de sortie à la prise de décision et les couches cachées aux différents niveaux de traitement de l'information visuelle.

### 5.3. L'algorithme de la rétropropagation

Plusieurs algorithmes ont été proposés pour l'apprentissage supervisé des poids synaptiques d'un réseau multicouche. La rétropropagation du signal d'erreur est l'algorithme le plus utilisé, sans doute grâce aux résultats obtenus avec cet algorithme. Désigné couramment en anglais par le terme *« backpropagation »*, il est une généralisation de l'algorithme de Widrow-Hoff pour un réseau multicouche. Il a d'abord été mis au point par [Werbos, 1974]dans le cadre de sa thèse de doctorat, et donc faiblement diffusé dans la communauté scientifique qui n'y a pas porté attention en cette période où la recherche en Intelligence Artificielle était surtout orientée vers la paradigme symbolique.

L'algorithme de rétropropagation du signal d'erreur a par la suite été redécouvert simultanément et indépendamment par [ le Cun, 1985], et [ Rumelhart, Hinton, & Williams, 1986].

#### 5.3.1. Fonction de sortie

Une particularité de cet algorithme consiste à utiliser une fonction non linéaire du type sigmoïde au lieu de la fonction seuil utilisée dans le modèle de *McCulloch&Pitts*. Cela a pour avantage de faciliter le calcul des différentes dérivées associées à l'évaluation des facteurs de

certain poids durant la phase rétropropagation sans toutefois apporter de grandes modifications au modèle de base du neurone formel adaptatif.

La fonction sigmoïde la plus souvent utilisée a pour expression :

$$f(a)=\frac{1}{1+e^{-\sigma a}} \qquad (2)$$

Avec a : la valeur d'activation du neurone et

$\sigma$ : Le facteur de pente de la sigmoïde

Plus la valeurs « $\sigma$ » est grande, plus la fonction sigmoïde s'approche de la fonction seuil.

La dérivée de la fonction de sortie du neurone est nécessaire au calcul du gradient. La dérivée de la fonction sigmoïde devient très simple à calculer lorsqu'on se rappelle que la fonction exponentielle est la seule fonction dont la dérivée est égale à elle-même. Des manipulations simples permettent d'exprimer la dérivée de la sigmoïde comme une fonction de la sortie seulement :

$$y=f(a)=\frac{1}{1+e^{-\sigma a}}$$

$$y'=f'(a)=\sigma y(1-y) \qquad (3)$$

### 5.3.2. Base d'apprentissage

Pour réaliser l'apprentissage des poids synaptiques d'un réseau perceptron multicouche, on dispose d'une base d'apprentissage comportant K Couples :

$$B=\{(X_k,D_k,k=1,2,.....K)\}$$

Où $X_k=[x_1(k), x_2(k),..... X_n(k),..... x_N(k)] \in \Re^N$ avec k=1,2,...,K, une des formes présentées à l'entrée, N est la dimension du vecteur d'entrée,

$D_k = (d_1(k), d_2(k),..... d_m(k),..... d_M(k)) \in \{0,1\}^M$ est le vecteur de sortie désirée correspondant à $X_k$, et M représente le nombre de classes à discriminer.

### 4.3.3. Architecture

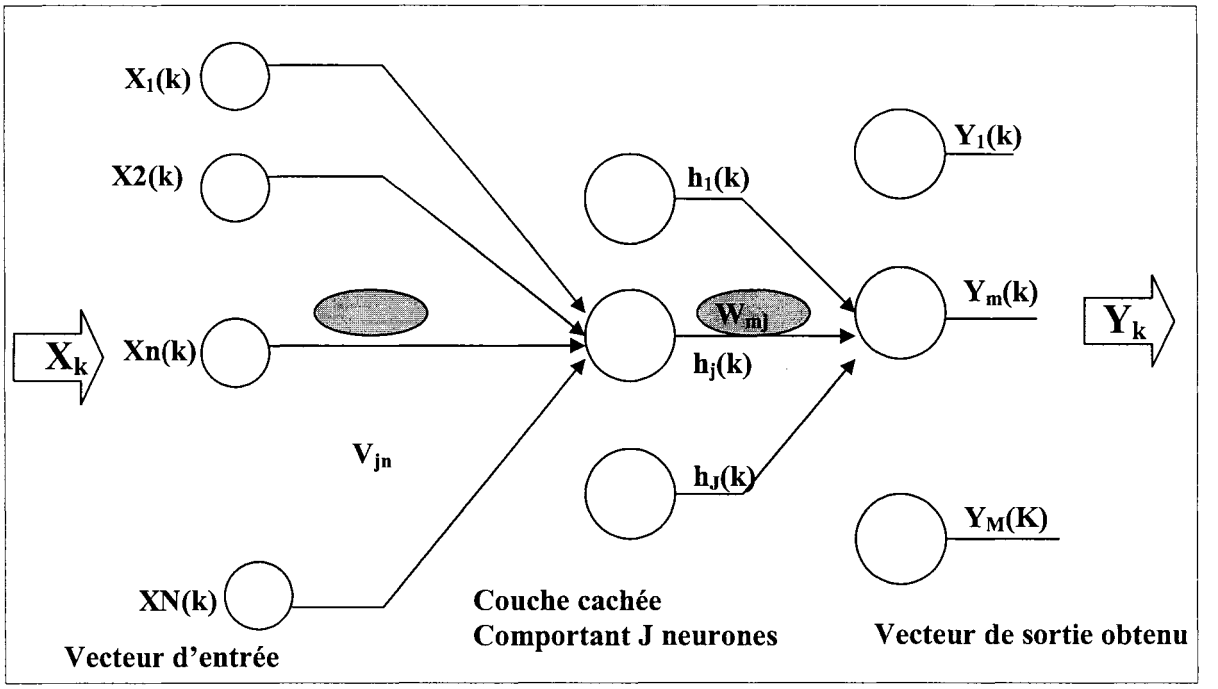



**Figure II.7** Architecture d'un réseau perceptron multicouche avec une couche cachée.
*Chaque neurone d'une couche est connecté à tous les neurones de la couche précédente.*

Posons $v_{jn}$ le poids synaptique de pondération de la connexion entre le neurone « n » de la couche d'entrée (n=1,2,……,N) et le neurone « j » de la couche cachée (j=1,2,….,j), où J est le nombre de neurones utilisés dans cette couche.

De même, posons $w_{mj}$ le poids synaptique de pondération de la connexion entre le neurone « j » de la couche cachée et le neurone « m » de la couche de sortie (m=1,2,….,M). Notons à ce niveau que seul le nombre de neurones dans la couche cachée J et le paramètre de la pente de la fonction sigmoïde utilisée $\sigma$ sont à déterminer avant que ne débute la phase de l'apprentissage. N(i.e. la dimension des vecteurs d'entrée) et M (i.e. le nombre de classes à discriminer) sont imposés par le problème posé.

### 5.3.4. Propagation directe

La première phase d'opération du perceptron multicouche consiste à propager la forme d'entrée à classifier, $X_k$, jusqu'à la sortie du réseau. La forme d'entrée est d'abord propagée sur la couche cachée pour produire le vecteur $H_k$, qui est lui-même propagé par la suite sur la couche de sortie du réseau, $Y_k$.

La sortie du neurone $h_j$ de la couche cachée et du neurone $y_m$ de la couche de sortie est donnée par :

$$H_j(k) = f\left(\sum_{n=0}^{N} v_{jn} x_n(k)\right)$$

$$Ym(K) = f\left(\sum_{j=0}^{j} w_{mj} h_j(k)\right) = f\left(\sum_{j=0}^{J} w_{mj} f\left(\sum_{n=0}^{N} v_{jn} x_n(k)\right)\right) \qquad (4)$$

Avec $v_{J0} = \beta_J$ la valeur de polarisation du neurone caché $_j$,

$W_{m0} = \beta_m$ la valeur de polarisation du neurone de sortie $y_m$

$X_0(k) = h_0(k) = +1$: l'extension du vecteur avec une composante constante unitaire pour simuler la polarisation du neurone.

Chaque neurone de la couche cachée et de la couche de sortie est doté d'une connexion supplémentaire reliée à une source constante unitaire qui simule une valeur de polarisation distincte pour chacun des neurones. Cette valeur de polarisation offre un degré supplémentaire de liberté au neurone en permettant de déplacer selon l'axe horizontal la fonction de sortie du neurone (fonction sigmoïde en général, parfois la fonction linéaire pour la couche de sortie). Les valeurs de polarisation permettent globalement de déplacer par translation les courbes de séparation de classes dans l'hyper-espace de la sortie. Cette connexion supplémentaire est soumise à l'entraînement du réseau, au même titre que toutes les autres connexions synaptiques.

### 5.3.5. Entraînement - modification des poids synaptiques

La fonction de coût que l'on cherche à minimiser est celle de l'***erreur quadratique instantanée*** définie par :

$$E(k)=\frac{1}{2}\sum_{m=1}^{M} e_m^2(k)=\frac{1}{2}\sum_{m=1}^{M}\left(d_m(k)-y_m(k)\right)^2 \quad (5)$$

Avec $e_m(k) = d_m(k)-y_m(k)$ l'erreur instantanée à la sortie du neurone $m$ pour l'entrée $X_k$ présentée à l'entrée du réseau.

L'algorithme de minimisation utilisé est celui de la descente du gradient stochastique (Blayo & Verleysen, 1996). Le gradient d'une fonction en un point est défini comme le vecteur qui pointe vers le maximum local de cette fonction le long de la pente la plus abrupte. Une technique de minimisation de l'erreur quadratique instantanée selon le négatif du gradient assure donc convergence relativement rapide vers une erreur minimum (à tout le moins localement).

L'algorithme de descente de gradient stochastique consiste donc à exprimer le gradient en fonction des poids de connexion du réseau et à trouver l'amplitude et le sens des changements de poids qui minimisent le gradient de la fonction d'erreur instantanée pour la forme $X_k$ présentée à l'entrée du réseau.

### 5.3.6. Poids synaptiques de la couche de sortie : $w_{mj}$

Au départ, la seule source d'erreur quantifiable est l'erreur de sortie. Le réseau sera donc d'abord calculé pour chacun des neurones de la couche de sortie et exprimé en fonction du poids des connexions qui parviennent à chaque neurone. La modification de poids sera apportée dans la direction du négatif du gradient.

Le vecteur de l'erreur exprime en fonction du poids des connexions parvenant à la couche de sortie est un vecteur de $M \times J$ composantes qui pointe vers le maximum local de l'erreur quadratique instantanée.

Cette rétropropagation est nécessaire afin de calculer la contribution des neurones de la couche cachée à l'erreur total que l'on peut mesurer à la sortie du réseau.

### 5.3.7. Algorithme

L'algorithme complet de rétropropagation du signal d'erreur avec correction du poids des connexions selon le négatif du gradient de l'erreur est présenté ci-dessous. L'algorithme est basé sur la méthode d'apprentissage par l'exemple dans laquelle le poids des connexions est ajusté a chaque présentation d'une forme $X_K$ provenant de la base d'apprentissage [2].

**Algorithme de rétropropagation**

1- Initialisation poids $Wij^k$ par des petites valeurs aléatoires ;

$$W_{ij}^{k} = \text{Random}$$

2- Présentation de la sortie désirée.
3- Présentation d'un exemple à l'entrée et calcul de la sortie de chaque couche et l'erreur correspondante.
4- Calcule des dérivées partielles par rapport à chaque poids, et adaptation des poids.
5- Retour à « 3 » et arrêt du processus si les sorties sont suffisamment proches des sorties désirées.

## 6. Conclusion

Le grand avantage des réseaux de neurones réside dans leur capacité d'apprentissage automatique, ce qui permet de résoudre des problèmes sans nécessiter l'écriture de règles complexes, tout en étant tolérant aux erreurs. Cependant, ce sont de véritables boites noires qui ne permettent pas d'interpréter les modèles construits. En cas, d'erreurs du système, il est quasiment impossible d'en déterminer la cause.

# CHAPITRE 4

# - Résultats et interprétations -

# 1. Introduction

Notre travail est composé de deux parties :

La première partie consiste à classifier des données mathématiques afin de retirer une conclusion vis-à-vis de la matrice finale qui sera utilisée comme entrée pour l'algorithme de rétropropagation avec les deux modèles supervisé et non supervisé.

Cette conclusion, on va la supposer comme théorie qui dit que la phase principale parmi les différentes phases qui constituent la procédure du *Data Mining* est d'avoir une matrice bien préparée et qui est constituée de données qui s'adaptent facilement à l'algorithme choisi pour effectuer la classification.

La deuxième partie sera consacrée à la classification des données médicales afin de prouver notre théorie.

# 2. Première partie du travail

Concernant la première partie du notre travail consiste à évaluer la classification supervisée et celle non-supervisée en mettant en œuvre un classifieur des données mathématiques (CDMDG). La rétropropagation est un algorithme de réseau de neurones qui présente l'avantage de supporter les deux approches qui nous intéressent. Son implémentation nous permet de juger de l'efficacité du modèle supervisé et celui non supervisé sans que le choix de l'algorithme de base, son paramétrage, notre maitrise et notre degré de compréhension de ce dernier n'altèrent les résultats finaux.

## 2.1. Choix des données

Les réseaux de neurones, comme toutes les autres techniques de *Data Mining* supportent très mal les données brutes et contenant, par conséquent, des erreurs, des valeurs manquantes, des valeurs Nulles, des redondances et d'autres types d'incohérences. Notre souci était, donc, d'éviter à tout prix ce genre de problèmes pour ne se focaliser que sur notre objectif. Pour ce faire, nous avons adopté une stratégie basée su trois critères essentiels :

- Le premier critère est le choix des données et leur taille. Les expériences rapportées dans la documentation scientifique, nous démontrent clairement que plus les données de la phase d'apprentissage sont volumineuses et bien préparées, meilleure est l'efficacité de la technique et

plus probante est la qualité des résultats [4]. Notre choix de données mathématiques synthétisées facilite grandement le respect de ce premier critère. Plus les données sont de qualité et présentent une grande similitude intra classe et assez de disparité et dissemblance interclasse, moins, on a besoin de données pour un apprentissage efficace de notre réseau de neurones [4]. Notre base est composée de 1200 vecteurs répartis sur 4 classes, dont chacune contient un même nombre de 300 éléments. En considérant la grande qualité de nos données, le nombre réduit de classes, le nombre d'éléments par classe, il nous a paru évident que les 1200 données sus-mentionnés sont amplement suffisantes pour effectuer convenablement la classification.

- Le deuxième critère est la qualité des données qui reste un critère fortement lié à celui de la taille. Nous avons choisi des données mathématiques. Ces données sont constituées d'un ensemble de vecteurs qui contient quatre classes en deux dimensions.

- Le troisième critère est la bonne préparation des données qui reste une étape très importante du processus de Data Mining mais malheureusement très peu étudiée pour ne pas dire ignorée par la recherche scientifique. Sachant que la préparation des données se fait en fonction de l'usage qu'on veut en faire et surtout de la technique utilisée, nous avions à choisir des données facilement utilisable par le modèle de rétropropagation. Notre choix des données mathématiques synthétisées répondait exactement au besoin de l'algorithme utilisé. Il nous permettait aussi d'éviter la préparation des données proprement dite. Soit, nous n'avions plus à traiter des problèmes tels que l'élimination des valeurs nulles, ou manquantes, le traitement des erreurs et des redondances et d'autres plus subtiles comme la surreprésentation d'une classe ou la forte corrélation entre les données.

Nous avons donc choisi un ensemble de données artificielles, générées à partir d'un programme C++ de telle façon que les classes dans chaque ensemble suivent une distribution gaussienne. Cette distribution nous permet de calculer la covariance de chaque classe afin de savoir le lien entre les différentes valeurs qui constituent chaque classe. Ce lien, est-ce que est faible ou fort, et ça sa dépend de la valeur de le covariance. Si ce lien est faible, on peut dire que les deux variables sont alors dites indépendantes, si non (le lien est fort), les deux variables sont alors dites dépendantes.

Dans notre cas, on n'est pas intéressé par ce calcul, on va juste profiter de ces différentes classes qu'on a obtenues pour les classifier en utilisant l'algorithme de rétropropagation avec les deux modèle supervisé et non supervisé, afin de retirer une comparaison à la fin du notre travail.

### 2.2. Classification

Soit la représentation d'un objet quelconque au moyen d'un vecteur de caractéristiques X=[x1, x2,....., xd]. Tous les vecteurs qui représentent l'ensemble des objets peuvent être positionnés dans l'espace euclidien $R^d$, où ils correspondent chacun à un point. Ceux-ci peuvent alors être regroupés en amas, chacun de ces amas étant associé à une classe particulière.

La classification, dans notre cas, a comme objectif de tester l'efficacité de l'algorithme de la rétropropagation en comparant les deux approches qu'il peut supporter, soit l'approche supervisée et non supervisée.

### 2.3. Principe général de l'algorithme

La rétropagation est l'algorithme le plus utilisé, sans doute grâce aux résultats obtenus avec cet algorithme.

Une particularité de cet algorithme consiste à utiliser une fonction du type sigmoïde, cela a pour avantage de faciliter le calcul des différentes dérivées.

La fonction sigmoïde la plus souvent utilisée a pour expression : $f(a) = \frac{1}{1+e^{-\partial a}}$

avec a : valeur d'activation

$\partial$ : facteur de pente de la sigmoïde.

### 2.4. Propagation

La première phase d'opération du perceptron multicouche consiste à propager la forme d'entrée à classifier, $X_k$, jusqu'à la sortie du réseau. La forme d'entrée est d'abord propagée sur la couche cachée pour produire le vecteur $H_k$, qui est lui-même propagé par la suite sur la couche de sortie du réseau, $Y_k$.

La sortie du neurone $h_j$ de la couche cachée et du neurone $y_m$ de la couche de sortie est donnée présentée dans le chapitre II, paragraphe 4.3.2 et 4.3.3.

## 2.5. Le modèle supervisé

Pour réaliser l'apprentissage des poids synaptiques d'un réseau perceptron multicouche, on dispose d'une base d'apprentissage comportant K couples :

$$B=\{(X_k,D_k,k=1,2,.....K)\}$$

Où $X_k=[x_1(k), x_2(k),..... X_n(k),..... x_N(k)] \in \Re^N$ avec k=1,2,...,k, une des formes présentées à l'entrée, N est la dimension du vecteur d'entrée,

$D_k = (d_1(k), d_2(k),..... d_m(k),..... d_M(k)) \in \{0,1\}^M$ est le vecteur de sortie désirée correspondant à $X_k$, et M représente le nombre de classes à discriminer.

La classification des données avec le modèle supervisé se fait en plusieurs étapes :

**Choix de l'échantillon d'apprentissage :** Comme cité ci-dessus, notre base est composée de 1200 vecteurs, de dimensions 2, représentant 4 classes. Chacune des classes est constituée de 300 vecteurs. Il est recommandé dans la littérature, selon la dimension de la base d'expérimentation, le nombre de classes, la répartition des données dans les classes et surtout le type d'algorithme de classification choisi, de prendre entre 20 et 80% des données pour l'apprentissage [4]. Puisque nos données sont d'une très grande qualité, il nous a semblé judicieux de ne considérer que 25% des éléments de chaque classe pour entreprendre l'apprentissage du réseau. Si, toutefois, ce seuil minimal ne nous permet pas de réaliser des résultats satisfaisants, nous aurons à reconsidérer la taille de l'échantillon d'apprentissage.

**Détermination du vecteur de sortie :** Le vecteur de sortie dépend uniquement du nombre de classe de la base de données. Comme nous avions 4 classes distinctes, nous avions à déterminer 4 valeurs de sortie correspondant chacune à une classe. Notre choix s'est porté sur 4 valeurs simples que sont {-1, 0, 1, 2}. On peut toujours, lors d'expérimentation future, mettre en cause notre choix en fonction des résultats.

**Création du réseau :** Après avoir choisi les données pour la phase d'apprentissage, la tâche suivante consiste en la création du réseau de telle façon que ce dernier puisse distinguer les différentes classes. À notre connaissance, il n'existe aucune méthode scientifique éprouvée qui permettre la création d'un réseau de façon automatique ou du moins systématique. L'approche la plus simple et la plus évidente est de choisir, de façon provisoire, un réseau initial. Selon les résultats obtenus, on effectue des changements au niveau du nombre de couches et du nombre de

neurones qui les constituent. D'après nos expérimentations, le réseau qui a donné les meilleurs résultats est un réseau de trois couches tel qu'il est représenté dans la figure III.1. La couche d'entrée est constituée de quinze neurones, la couche cachée possède deux neurones, alors que la couche de sortie n'est composée que d'un seul neurone.

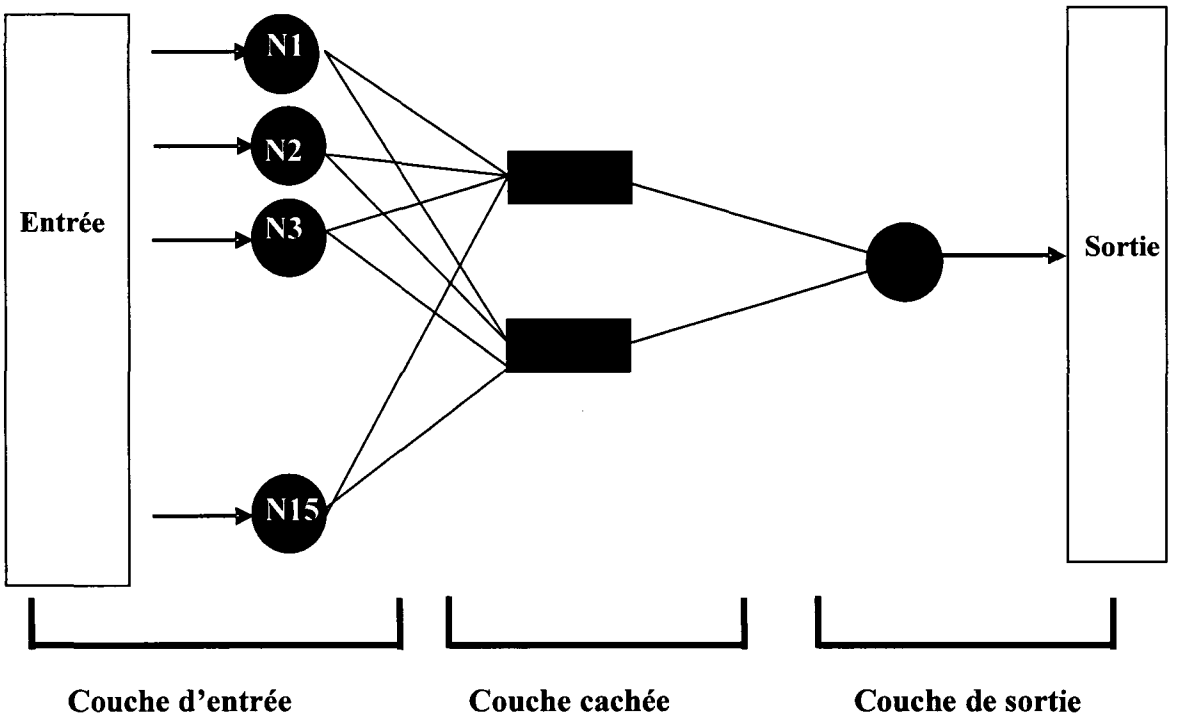



**Figure III.1.** Architecture du réseau pour le modèle supervisé

**Apprentissage :** Il consiste à calculer les différentes valeurs de sortie en fonction des différents paramètres d'entrée, soit, le réseau lui-même, le vecteur d'entrée qui est constitué des données d'apprentissage et le vecteur de sortie. L'algorithme va tenter lui-même, sur plusieurs passages d'aboutir au vecteur de sortie en faisant les corrections nécessaires selon les valeurs de sortie obtenues et celles désirées. Dans notre cas, le nombre d'itérations pour aboutir aux résultats voulus, était minime, puisqu'il nous a suffi d'environ soixante dix itérations pour que l'algorithme converge, comme il est illustré au figure III.02.

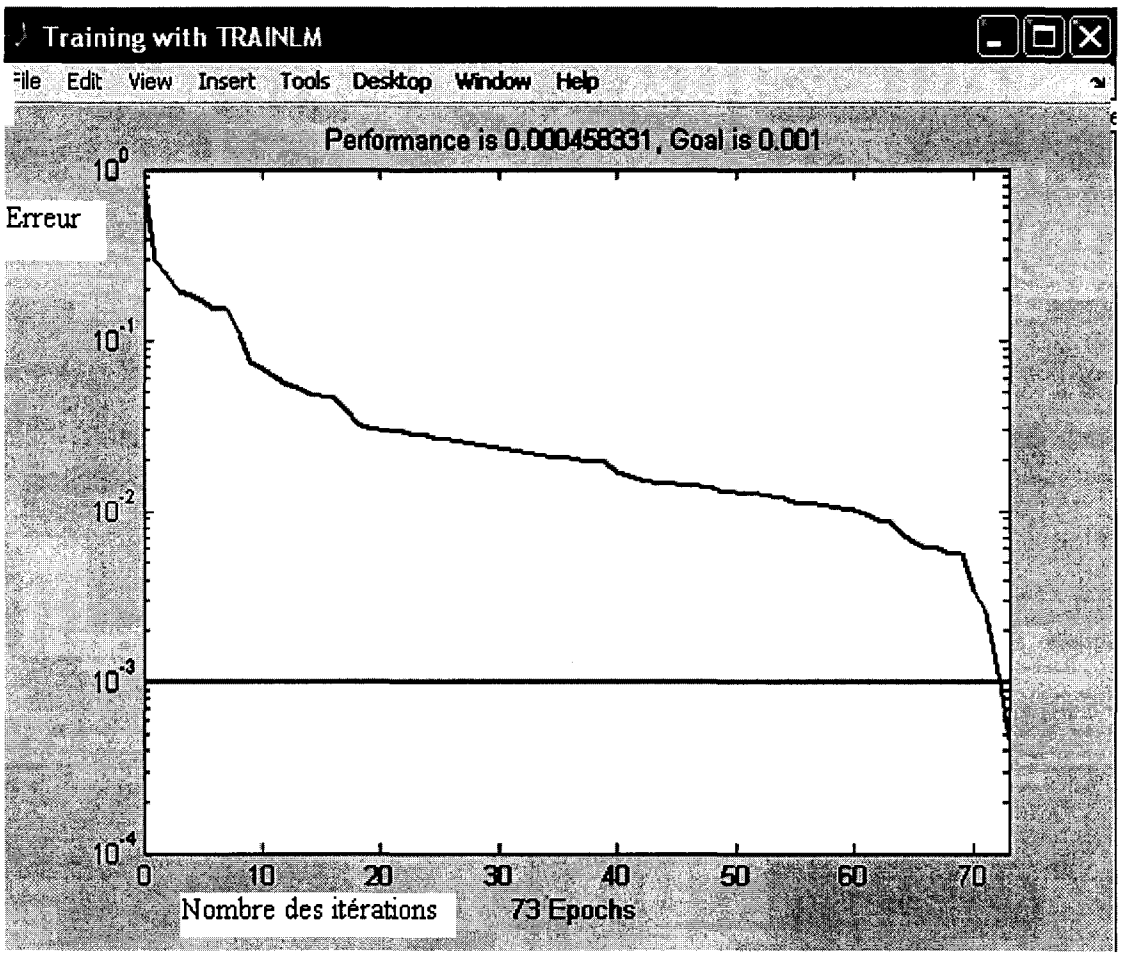



**Figure III. 02 :** Graphe de convergence du modèle supervisé lors de l'apprentissage

Le réseau calcule les valeurs de la couche de sortie en fonction des vecteurs d'entrée. L'erreur est la différence entre la valeur de sortie calculée par le réseau et la valeur désirée. À l'itération suivante, l'algorithme ajuste les poids synaptiques dans le but de converger vers les bonnes

valeurs. L'algorithme s'arrête si l'erreur calculée est inférieure à l'erreur établie par l'utilisateur. Dans nos expérimentations la condition d'arrêt était définie par Erreur < 0.001. D'après la figue III.02, on constate que le nombre d'itérations pour satisfaire la condition d'arrêt était de 73.

| *Classes* | | **Valeurs d'entrée et de sortie pour l'apprentissage** | | | | |
|---|---|---|---|---|---|---|
| *C1* | *E* | (0.268603,1.180803) | (-0.13398, 1.180803) | (0.259981,-0.399098) | (0.107221,0.177354) | (0.034734,1.107159) |
| | *S* | **-0.9902** | **-1.0223** | **-0.9901** | **-0.9722** | **-1.0227** |
| | *E* | (-0.771129,0.211766) | (0.887135,-0.186182) | (0.050308,1.181539) | (-0.475199, 0.436368) | (0.037424,-0.593919) |
| | *S* | **-0.9997** | **-1.0264** | **-0.9901** | **-1.0184** | **-0.9642** |
| *C2* | *E* | 1.37803,0.539227) | (2.025658,-0.402426) | (2.506219,0.969753) | (1.706382,-0.030229) | (1.909551,0.999828) |
| | *S* | **-0.0229** | **-0.0016** | **-0.0020** | **-0.0124** | **-0.0279** |
| | *E* | (1.909551,0.999828) | (2.2425,-0.03856) | (2.480621,-0.261409) | (2.016677, -0.737001) | (1.978694,-0.490305) |
| | *S* | **-0.0023** | **-0.0302** | **-0.0070** | **-0.0085** | **0.0062** |
| *C3* | *E* | (-0.133293,1.373162) | (0.733563,1.983601) | (-0.81891,3.842087) | (0.218072, 3.068039) | (1.294038,3.677364) |
| | *S* | **0.9972** | **0.9986** | **0.9901** | **0.9968** | **0.9978** |
| | *E* | (-0.252656,3.524044) | (0.2327,2.799661) | (0.060446,3.028009) | (1.549318, 1.547245) | (-0.349206,3.95048) |
| | *S* | **1.0005** | **1.0142** | **1.0273** | **1.0066** | **1.0120** |
| *C4* | *E* | (2.499991,3.026773) | (3.840283,1.641172) | (2.853216,2.057758) | (3.058334, 2.713721) | (3.84125,3.13484) |
| | *S* | **1.9831** | **1.9832** | **1.9689** | **1.9830** | **1.9829** |
| | *E* | (2.218216,1.429079) | (3.137477,1.809448) | (3.09368,2.006414) | (3.144071,1.958012) | (3.144071,1.958012) |
| | *S* | **2.0390** | **1.9551** | **1.9301** | **2.0397** | **2.0097** |

**Ci** : représente la ième classe
**E** : représente le couple d'entrée
**S** : représente la valeur de sortie

**Tableau III. 03 : Quelques valeurs de sortie lors de l'apprentissage du modèle supervisé selon l'algorithme de rétropropagation**

Le tableau III.03 représente quelques valeurs des couples d'entrée qu'on a pris pour la phase d'apprentissage et les différentes valeurs de sorties. On constate que toutes les sorties qui ont la valeur -1 ou proche d'elle, elles appartient à la classe 1, qui ont la valeur 0 ou proche d'elle, elles appartient à la classe 2, qui ont la valeur 1 ou proche d'elle, elles appartient à la classe 3, qui ont la valeur 2 ou proche d'elle, elles appartient à la classe 4.

**Tests et validation :** la validation du modèle issu de l'apprentissage consiste en une série de tests. Comme expliqué précédemment, nous avons utilisé un échantillon de 25% des données

pour l'apprentissage. Les 75% des données restantes peuvent être utilisées pour tester la validité de l'algorithme.

| *Classes* | | **Valeurs d'entrée et de sortie pour le test** | | | | |
|---|---|---|---|---|---|---|
| *C1* | *E* | (0.092209,-0.449112) | (0.373242,-1.302215) | (0.028342,0.485813) | (1.120097,0.642008) | (-0.206956,-0.123313) |
| | *S* | **-1.0147** | **-1.0248** | **-1.0191** | **-1.0250** | **-0.9632** |
| | *E* | (-0.560191,-0.325894) | (0.862126,1.549547) | (0.459002,-0.363115) | (-0.224402,0.390621) | (-0.310595,0.083764) |
| | *S* | **-1.0242** | **-1.0174** | **-1.0120** | **-1.0271** | **-1.0242** |
| *C2* | *E* | (2.499509,0.294501) | (2.015032,-1.108431) | (2.492354,0.519921) | (3.531698,0.546356) | (3.946457,-1.043143) |
| | *S* | **-0.0260** | **-0.0213** | **-0.0448** | **-0.0150** | **-0.0063** |
| | *E* | (3.874392,-0.378449) | (1.316915,-0.23079) | (2.782012,0.222345) | (1.685263,0.481936) | (3.111397,0.914298) |
| | *S* | **-0.0164** | **-0.0243** | **-0.0094** | **-0.0274** | **-0.0036** |
| *C3* | *E* | (1.654163,3.252157) | (-0.323065,3.199364) | (0.678281,2.729001) | (0.476487,2.925513) | (0.019909,2.540219) |
| | *S* | **0.9961** | **1.0007** | **1.0191** | **1.0123** | **0.9951** |
| | *E* | (0.022347,2.899765) | (-0.207373,2.807434) | (0.695322,4.267068) | (0.4916,1.685123) | (0.272533,3.119565) |
| | *S* | **1.0188** | **0.9974** | **1.0114** | **0.9892** | **0.9949** |
| *C4* | *E* | (0.981995,2.975464) | (0.521567,2.453377) | (0.815571,3.447889) | (0.907462,3.027059) | (0.684725,3.428794) |
| | *S* | **1.9679** | **2.0262** | **1.9832** | **1.9831** | **1.9850** |
| | *E* | (0.836894,3.496316) | (0.47724 ,3.346581) | (0.190854,2.645528) | (1.599451,2.304157) | (1.210517,2.448392) |
| | *S* | **2.0097** | **1.9999** | **2.0389** | **2.0376** | **2.0099** |

**Ci** : représente la ième classe
**E** : représente le couple d'entrée
**S** : représente la valeur de sortie

**Tableau III. 04 : Quelques valeurs de tests de validation du modèle supervisé selon l'algorithme de rétropropagation**

La même explication qu'on a déjà présentée précédemment dans le tableau III.03, pour les différentes valeurs.

Les quelques individus représentés dans le tableau 04 sont issus des 75% des données laissées pour le test. Nous constatons que quelque soit la classe à laquelle appartiennent les données, la classification s'est effectuée de façon adéquate. Dans le tableau 05 sont recensés les valeurs minimales et maximales des sorties obtenues par le réseau pour les quatre classes, ainsi que les marges d'erreurs possibles. Il est aisé de constater que le taux de classification est de 100% avec des erreurs de valeurs de sorties minimes, voire négligeables.

| *Classes* | *Valeur de sortie* | *Max* | *Min* | *Δ* |
|---|---|---|---|---|
| 1 | -1 | -0.9632 | -1.0250 | -0.0618 |
| 2 | 0 | -0.0448 | -0.0016 | -0.0432 |
| 3 | 1 | 1.0191 | 0.9949 | 0.0242 |
| 4 | 2 | 2.0262 | 1.8472 | 0.179 |

**Tableau III. 05 :** Les valeurs minimales et maximales des sorties pour le modèle supervisé

| *Classes réelles* | *Classes obtenues* | | | |
|---|---|---|---|---|
| | **1** | **2** | **3** | **4** |
| **1** | 100% | 0 | 0 | 0 |
| **2** | 0 | 100% | 0 | 0 |
| **3** | 0 | 0 | 100% | |
| **4** | 0 | 0 | 0 | 100% |

**Tableau III. 06 :** Le taux de classification pour le modèle supervisé

### 2.6. Le modèle non supervisé

Les réseaux, utilisant l'apprentissage non supervisé, sont souvent appelés auto-organisateurs, ou encore à apprentissage compétitif. Dans ce type d'apprentissage la connaissance de la sortie désirée n'est pas nécessaire, c'est à dire que le réseau s'*auto-organise* et *organise* les entrées qui sont présentées comme vecteur d'entrée.

**Choix de l'échantillon d'apprentissage :** L'approche du choix de l'échantillon pour le modèle non supervisé est exactement la même que le choix dans le précédent cas du modèle supervisé. On n'a alors considéré que 25% des éléments de chaque classe pour entreprendre l'apprentissage du réseau. Nous nous réservons le droit de reconsidérer la taille de l'échantillon d'apprentissage si les résultats obtenus ne sont pas satisfaisants.

**Détermination du vecteur de sortie :** Dans ce type d'apprentissage la connaissance de la sortie désirée n'est pas nécessaire. Le réseau aura la tâche d'organiser les données d'entrée afin de

produire les résultats les plus adéquats. Il incombe à lui, donc, de déterminer le nombre de classe de sortie, les valeurs de sorties pour chaque classe, ainsi que la classification de chaque élément.

**Création du réseau :** Le principe de création du réseau est le même que dans le modèle supervisé. Bien que le nombre de classes n'est pas connu d'avance, l'approche de création reste identique et consiste en une série d'expérimentations jusqu'à l'obtention de résultats satisfaisants. Après plusieurs tests, nous avons abouti à un réseau de trois couches tel qu'il est représenté dans la figure III.5. La couche d'entrée est constituée de dix neurones, la couche cachée possède deux neurones, alors que la couche de sortie n'est composée que d'un seul neurone.

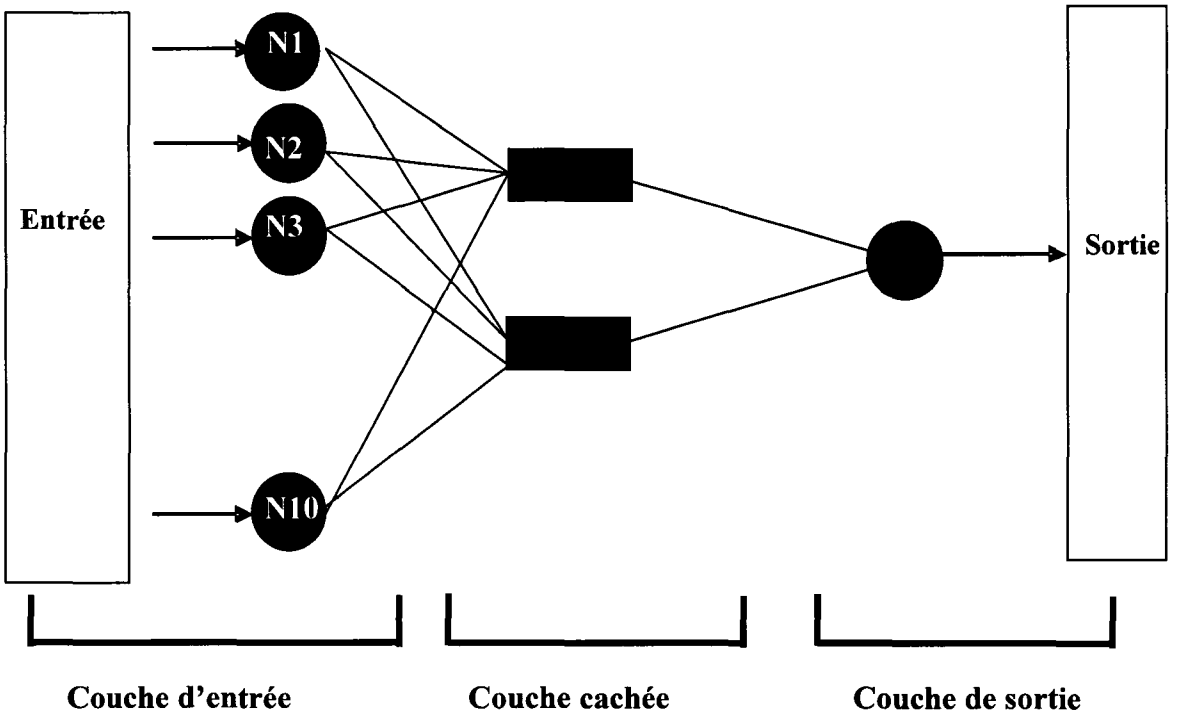



**Figure III. 7.** Architecture du réseau pour le modèle non supervisé

**Apprentissage :** La seule différence qui existe entre le modèle supervisé et celui non supervisé lors de l'apprentissage est que ce dernier n'a pas besoin de valeurs de sortie désirées. Il attribuera lui-même à chaque élément en entrée une valeur de sortie. En comparant les différentes valeurs

obtenues, nous pouvons vérifier si le réseau a bien dissocié les éléments des quatre classes. L'algorithme va tenter lui-même, sur plusieurs passages de regrouper les éléments similaires en entrée dans une même classe. Le nombre d'itération pour regrouper les données d'apprentissage était très faible et se limitait à quatre. Cette vitesse de convergence de l'algorithme peut facilement s'expliquer par la qualité des données. Nous rappelons que ce sont des données mathématiques synthétisées, sans aucune erreur ni omission. Aussi, les éléments des quatre classes sont très distinctifs les uns des autres et se prêtent favorablement à la classification. La figure III. 08 représente le graphe de convergence du réseau lors de l'apprentissage.

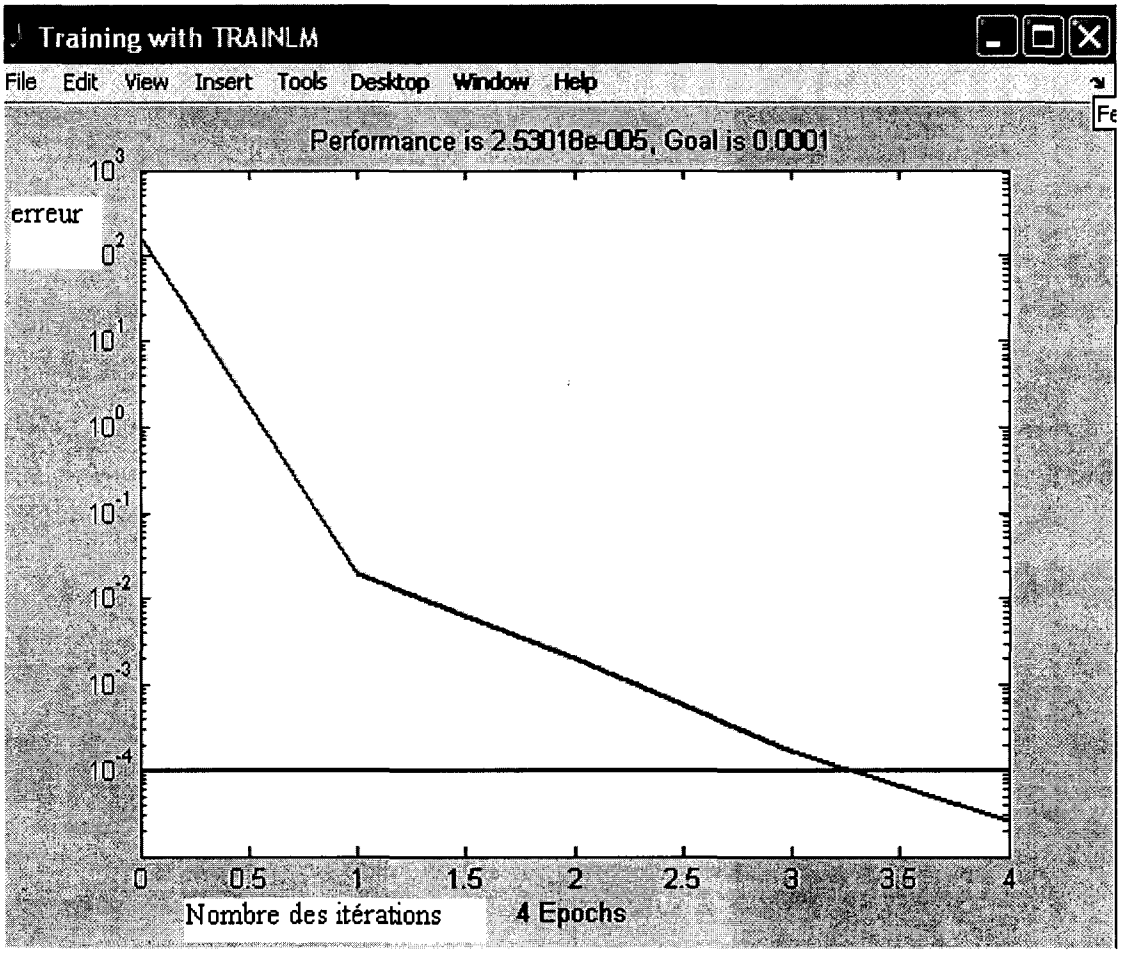



**Figure III. 08 :** Graphe de convergence du modèle non supervisé lors de l'apprentissage

| *Classes* | | Valeurs d'entrée et de sortie pour l'apprentissage | | | | |
|---|---|---|---|---|---|---|
| *C1* | *E* | (0.493088,-0.036919) | (0.775273,-1.524021) | (-0.358564,0.025118) | (0.80448,-0.309862) | (0.135086,0.871789) |
| | *S* | **0.299** | **0.2367** | **0.2782** | **0.2148** | **0.2255** |
| | *E* | (0.084217,-0.685947) | (0.075108,0.335772) | (-0.114131,-1.946092) | (-0.20523,-0.64263) | (-0.732188,0.604921) |
| | *S* | **0.2568** | **0.2043** | **0.2541** | **0.2861** | **0.2139** |
| *C2* | *E* | (2.099906,-0.90553) | (2.711258,-0.919658) | (2.570386,-0.623812) | (1.9779,-0.019775) | (2.482667,0.965654) |
| | *S* | **0.3008** | **0.3669** | **0.3732** | **0.3804** | **0.3122** |
| | *E* | (2.189983,0.297769) | (2.908564,0.950395) | (2.817401,-0.738229) | (2.660407,-0.225938) | (2.54579,-0.073514) |
| | *S* | **0.3148** | **0.3959** | **0.3376** | **0.3221** | **0.3347** |
| *C3* | *E* | (0.733563,1.983601) | (-0.81891,3.842087) | (0.218072,3.068039) | (1.294038,3.677364) | (-0.252656,3.524044) |
| | *S* | **0.0393** | **0.0603** | **0.0552** | **0.0599** | **0.01074** |
| | *E* | (0.2327,2.799661) | (0.060446,3.028009) | (1.549318,1.547245) | (-0.349206,3.95048) | (1.985534,3.685453) |
| | *S* | **0.0550** | **0.0556** | **0.0815** | **0.0683** | **0.01347** |
| *C4* | *E* | (2.218304,3.061643) | (2.631389,3.471232) | (2.80025,2.750623) | (3.675892,4.08345) | (3.393931,4.08345) |
| | *S* | **0.1917** | **0.1691** | **0.1418** | **0.1949** | **0.1988** |
| | *E* | (3.393931,3.385473) | (2.245022,3.040353) | (2.663725,2.70664) | (3.529525,2.919637) | (2.749239,2.211749) |
| | *S* | **0.1668** | **0.1375** | **0.1116** | **0.1288** | **0.1878** |

**Ci** : représente la ième classe
**E** : représente le couple d'entrée
**S** : représente la valeur de sortie

**Tableau III. 09 : Quelques valeurs de sortie lors de l'apprentissage du modèle non supervisé selon l'algorithme de rétropropagation**

La même explication qu'on a déjà présentée précédemment dans le tableau III.03, pour les différentes valeurs.

**Tests et validation :** De la même manière qu'effectuée pour le modèle supervisé, le test et la validation du réseau, issu de l'apprentissage, consiste à utiliser les 75% des données restantes pour vérifier la véracité de l'algorithme sur des données autres que celles qui ont servi à l'apprentissage.

| *Classes* | | Valeurs d'entrée et de sortie pour le test | | | | |
|---|---|---|---|---|---|---|
| *C1* | *E* | (0.862126,1.549547) | (0.459002,-0.363115) | (-0.224402,0.390621) | (-0.310595,0.083764) | (-0.212201,0.615746) |
| | *S* | **0.2692** | **0.2070** | **0.2681** | **0.2887** | **0.2311** |
| | *E* | (0.001577,0.764121) | (0.313069,-0.153178) | (-0.464378,-0.016926) | (-0.692192,-0.251436) | (0.29343,1.241986) |
| | *S* | **0.2077** | **0.2320** | **0.2903** | **0.2172** | **0.2385** |
| *C2* | *E* | (2.959524,0.07463) | (2.862529,-0.769952) | (2.685552,1.200155) | (2.879845,-0.393537) | (2.233539,-0.16403) |
| | *S* | **0.352** | **0.3120** | **0.3194** | **0.3460** | **0.3090** |
| | *E* | (2.687454,0.08279) | (2.499509,0.294501) | (2.015032,-1.108431) | (2.492354,0.519921) | (3.531698,0.546356) |
| | *S* | **0.3833** | **0.3296** | **0.3008** | **0.3669** | **0.3748** |
| *C3* | *E* | (0.208759,2.849819) | (0.736202,3.16161) | (0.78172,2.908616) | (0.65467,2.969138) | (0.917805,2.508661) |
| | *S* | **0.0606** | **0.0686** | **0.0899** | **0.1154** | **0.1063** |
| | *E* | (0.46098,1.606384) | (0.431532,2.842957) | (0.880959,2.205106) | (1.4080382.277327) | (1.388201,2.11712) |
| | *S* | **0.0372** | **0.0847** | **0.0659** | **0.0676** | **0.0832** |
| *C4* | *E* | (3.055938,2.221046) | (3.003407,3.185531) | (2.800097,3.307857) | (2.476742,2.246263) | (3.539988,3.046651) |
| | *S* | **0.1434** | **0.1634** | **0.1261** | **0.1061** | **0.1580** |
| | *E* | (3.090048,4.052457) | (3.115971,1.971713) | (3.235921,2.502946) | (3.227259,2.071381) | (3.672003,1.932905) |
| | *S* | **0.1490** | **0.1796** | **0.0870** | **0.0536** | **0.1211** |

**Ci** : représente la ième classe
**E** : représente le couple d'entrée
**S** : représente la valeur de sortie

**Tableau III. 10 : Quelques valeurs de tests de validation du modèle non supervisé selon l'algorithme de rétropropagation**

La même explication qu'on a déjà présentée pour le tableau III.03, pour les différentes valeurs.
Les quelques individus représentés dans le tableau III.10 sont issus des 75% des données laissées pour le test. On peut dire que, de manière générale, la classification s'est bien effectuée. Il existe, néanmoins, quelques éléments qui ont été mal classés. En ce qui concerne les classes 1 et 2, le taux de classification lors des tests est de 100%. Sachant que la limite supérieure des valeurs de sortie de la classe 1 est égale à $29*10^{-2}$ et la limite inférieure de la classe 2 est de $30*10^{-2}$, on s'attendait à ce que quelques éléments des deux classes soient mal affectés. Heureusement , les tests nous ont donnés tort. En ce qui concerne les classes 3 est 4, le taux de classification est de 83,33% dans les deux cas. 16,67% des éléments de la classe 3 se sont retrouvés dans la classes 4 et autant de cette même classe se sont retrouvés dans la classe 4. La limite supérieure des valeurs de sortie de la classe 3 étant égale à $8*10^{-2}$ et la limite inférieure de la classe 4 étant de $10*10^{-2}$,

nous soupçonnions qu'il ait quelques chevauchements des éléments des deux classes. Chose qui s'est confirmée après les tests effectués.

| *Classes réelles* | *Classes obtenues* | | | |
|---|---|---|---|---|
| | **1** | **2** | **3** | **4** |
| **1** | 100% | 0 | 0 | 0 |
| **2** | 0 | 100% | 0 | 0 |
| **3** | 0 | 0 | 83.33% | 16.67% |
| **4** | 0 | 0 | 16.67% | 83.33% |

**Tableau III. 11 :** Le taux de classification pour le modèle NON- supervisé

| *Classes* | *Valeur de sortie* | *Max* | *Min* | *Δ* |
|---|---|---|---|---|
| 1 | Non spécifiée | -0.9632 | -1.0271 | -0.0639 |
| 2 | Non spécifiée | -0.0016 | -0.0448 | -0.0432 |
| 3 | Non spécifiée | 1.0191 | 0.9949 | 0.0242 |
| 4 | Non spécifiée | 0.1796 | 0.0536 | 0.126 |

**Tableau III. 12 :** les valeurs minimales et maximales des sorties pour le modèle non-supervisé

## 2.7. Comparaison entre les deux modèles

La littérature scientifique est unanime sur l'efficacité de l'algorithme de la rétropropagation dans la classification [4], que ce soit pour le modèle supervisé ou non et quelque soit la nature et le type des données à classifier. Notre comparaison va plus porter sur la différence fondamentale entre les deux modèles, soit l'aspect supervisé ou non de l'apprentissage que sur la pertinence des résultats. Pour cela, nous avons divisé la comparaison sur cinq points essentiels, les données, l'algorithme, le modèle, l'apprentissage et la classification.

**Les données :**

Pour une comparaison plus juste et plus crédible, il était évident qu'il fallait opter pour les mêmes données. Un tel choix, nous permet d'éliminer une éventuelle influence des données sur les résultats produits par l'algorithme dans les deux cas. Notre choix de données synthétisées et de grande qualité réduit encore plus un quelconque impact négatif. L'absence d'erreurs, de redondances, de valeurs manquantes et de valeurs erronées atteste de la bonne qualité des données, qui d'une part facilite la tâche de la classification et rehausse la qualité des résultats et d'autre part permet une neutralité parfaite.

**L'algorithme :**

L'avantage de l'algorithme de la rétropropagation est qu'il peut servir à la classification pour les deux modèles supervisé et non supervisé. C'est cette principale caractéristique qui nous a amenés à opter pour ce type d'algorithme et va nous permettre, lors des comparaisons futures, de porter notre attention sur l'aspect de supervision ou non que sur l'algorithme lui même.

**L'architecture du modèle :**

Hormis le fait que dans le modèle supervisé, on a besoin de désigner un vecteur de sortie, alors que dans le modèle non supervisé, c'est l'algorithme lui-même qui cherche la sortie la plus appropriée, à même d'assurer les meilleurs résultats, les architectures des deux modèles restent sensiblement les mêmes. Pour les deux modèle, le réseau est constitué de trois couches, une d'entrée, une couche cachée et une autre de sortie. La seule différence réside dans le nombre de neurones de la couche d'entrée. Alors que dans le modèle supervisé le nombre de neurones qui nous a permis d'avoir des résultats optimaux était de quinze, celui-ci était de dix dans le modèle non supervisé. Pour la couche cachée et la couche de sortie, le nombre de neurones était, respectivement de deux et un dans les deux modèles. La différence minime de nombre de

neurones dans la couche d'entrée n'a, à notre connaissance, aucune signification ni explication scientifique. C'est les expérimentations que nous avons menées lors de l'apprentissage qui nous ont guidé, de façon empirique, vers ces choix d'architectures.

**L'apprentissage :**

La principale différence entre les deux modèles lors de l'apprentissage est le nombre d'itérations nécessaires à la convergence. Alors que dans le modèle supervisé, il a fallu à l'algorithme quelques 73 itérations pour converger vers la solution, ce dernier n'a eu besoin que de 4 itérations pour satisfaire les conditions de la convergence dans le cas du modèle non supervisé. Cette grande différence dans le nombre d'itérations s'explique par le choix que nous avons fait concernant les sorties désirées. On peut recenser deux facteurs majeurs qui ont accéléré la convergence du modèle non supervisé. Le premier est la grande qualité des données qui facilite la distinction entre les éléments des différentes classes. Le deuxième facteur est la faculté qu'a l'algorithme à s'auto-organiser selon les données d'entrée pour ajuster les sorties et produire les meilleurs résultats possibles. Dans le cas du modèle supervisé, nous pouvons justifier le relatif grand nombre d'itérations par l'obligation du réseau à aller chercher les sorties désirées et prédéfinies. C'est, donc, notre choix qui est à remettre en cause plutôt que l'algorithme lui-même. Il est utile de préciser que le nombre d'itérations que nous avons qualifié de grand, reste modeste et plus qu'acceptable si nous le comparons aux résultats de la littérature [Lin&Horne].

**La classification :**

Les résultats obtenus sont satisfaisants pour les deux modèles. Nous avons constaté, néanmoins, que le modèle supervisé est supérieur au modèle non supervisé puisqu'il a présenté un taux de classification de 100% alors que le modèle non supervisé s'est contenté de 83,33%. Le tableau III.13 résume les taux de classification des deux modèles en fonction des quatre classes.

| Classe | Taux de classification Modèle supervisé | Taux de classification Modèle non- supervisé |
|---|---|---|
| Classe1 | 100% | 100% |
| Classe2 | 100% | 100% |
| Classe3 | 100% | 83,33% |
| Classe4 | 100% | 83,33% |
| Total | 100% | 91,66% |

Tableau III13 : Le taux de classification pour les deux modèles supervisé et non supervisé

## 3. Deuxième partie du travail

La deuxième partie du notre travail est une partie d'expérimentation, elle sera consacrée complètement à classifier des données médicales.

### 3.1. Choix de données

Les données que nous avons choisi pour effectuer notre test de classification sont des données médicales réelles prises dans trois hôpitaux au Portugal [58].

Ces données représente un test rapide « Le score d'*APGAR* » qui permet d'évaluer l'état initial du nouveau-né, puis son évolution à une trois cinq dix minutes : Il permet aux médecins de déterminer la conduite à tenir ainsi que les éléments de surveillance de chaque nouveau-né.

Les résultats de cette évaluation en sont indiqués sur le carnet de santé de chaque nouveau-né.

Le score d' *APGAR* est un score utilisé pour évaluer la santé d'un nouveau-né à la naissance. Il est compris entre 0 (arrêt cardiaque) et 10 (normal). Il fut inventé par Virginia Apgar en 1952.

Notre matrice qu'on a préparée est composée d'un triplet, la première valeur représente le score d'*APGAR* pris après une minute de la naissance d'un bébé, la deuxième valeur représente le score d'*APGAR* pris après cinq minutes de la naissance d'un bébé et la troisième valeur est la duré dont laquelle le score d' *APGAR* reste fixe pendant le deuxième test.

### 3.2. Préparation de données

Notre base de données est composée de 228 vecteurs, de dimensions 3 représentent des données médicales des différents bébés dans trois hôpitaux au Portugal.

Concernant l'apprentissage, on va procéder le test sur 25% de données pour les deux modèles supervisé et non supervisé.

Voici la structure de la matrice qu'on a préparé pour effectuer la classification :

| **Nom de l'hôpital** | **Nom du bébé** | ***APGAR1*** | ***APGAR5*** | **Durée** |
|---|---|---|---|---|
| HUC | PMGVG | 9 | 10 | 44 |
| HUC | MCSR | 8 | 10 | 55 |
| . | . | | | |
| . | | | | |
| . | | | | |
| . | | | | |
| HGSA | RMSNP | 10 | 10 | 60 |
| HGSA | IRF | 10 | 10 | 40 |
| . | | | | |
| . | | | | |
| . | | | | |
| . | | | | |
| HSJ | CMB | 09 | 10 | 35 |
| HSJ | AMA | 8 | 9 | 30 |
| . | | | | |
| . | | | | |
| . | | | | |

### 3.3. Création du réseau

La création du réseau consiste en une série d'expérimentations jusqu'à l'obtention de résultats satisfaisants. On suit toujours le même principe que le précédent.

### 3.4. Le modèle supervisé

On va suivre les mêmes démarches que les expérimentations précédentes pour ce modèle. L'approche la plus simple et la plus évidente est de choisir, de façon provisoire, un réseau initial. Selon les résultats obtenus, on effectue des changements au niveau du nombre de couches et du nombre de neurones qui les constituent jusqu'à l'obtention des bons résultats.

Dans cette partie du notre travail, nous avons initialisé notre réseau avec quatre couches dont la première couche c'est la couche d'entrée, deuxième et troisième couche représentent les deux couches cachées et la dernière couche c'est la couche de sortie.

D'après nos expérimentations, le réseau tel qui représenté dans la figure III.10 a donné des meilleurs résultats, avec la couche d'entrée est constituée de quinze neurones, la première couche

cachée possède deux neurones, la deuxième couche cachée possède trois neurones, alors que la couche de sortie n'est composée que d'un seul neurone.

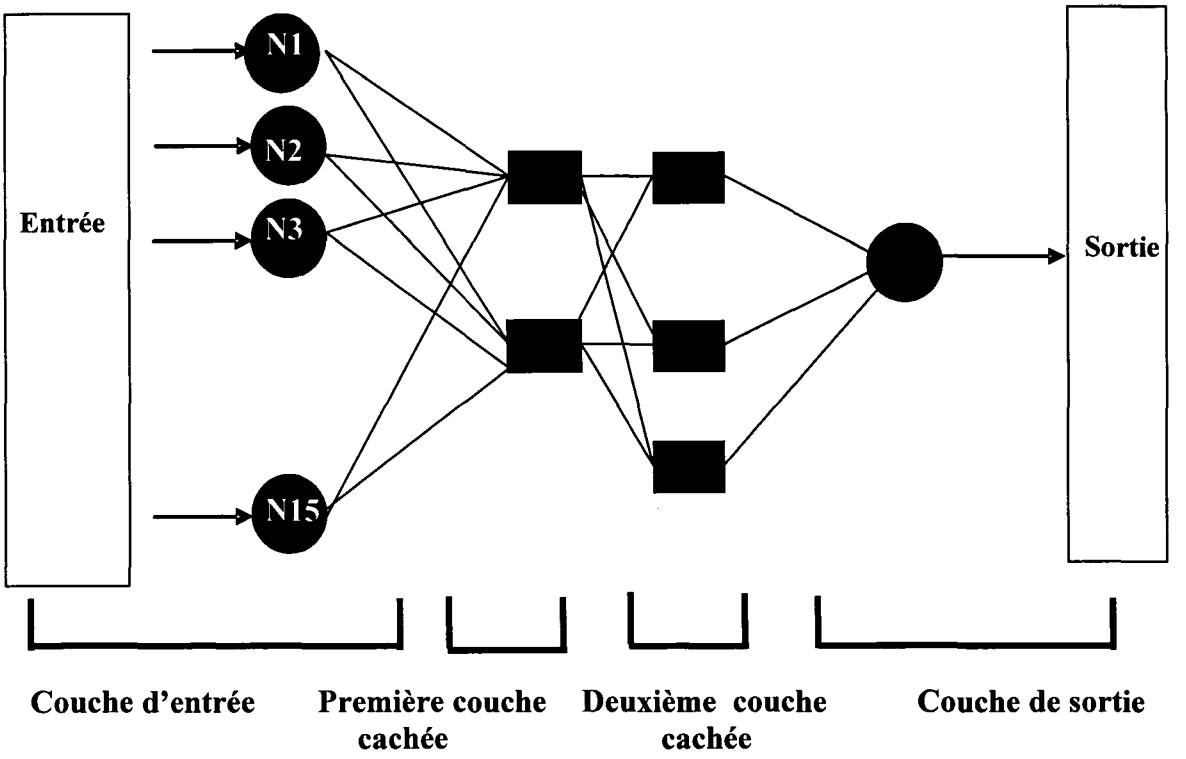



**Figure III.10.** Architecture du réseau pour le modèle supervisé

Concernant l'apprentissage pour la deuxième partie du notre travail, l'algorithme va tenter lui-même, sur plusieurs passages d'aboutir au vecteur de sortie en faisant les corrections nécessaires selon les valeurs de sortie obtenues et celles désirées. Dans notre cas, le nombre d'itérations est 358 pour que l'algorithme converge, comme il est illustré à la figure III.11.

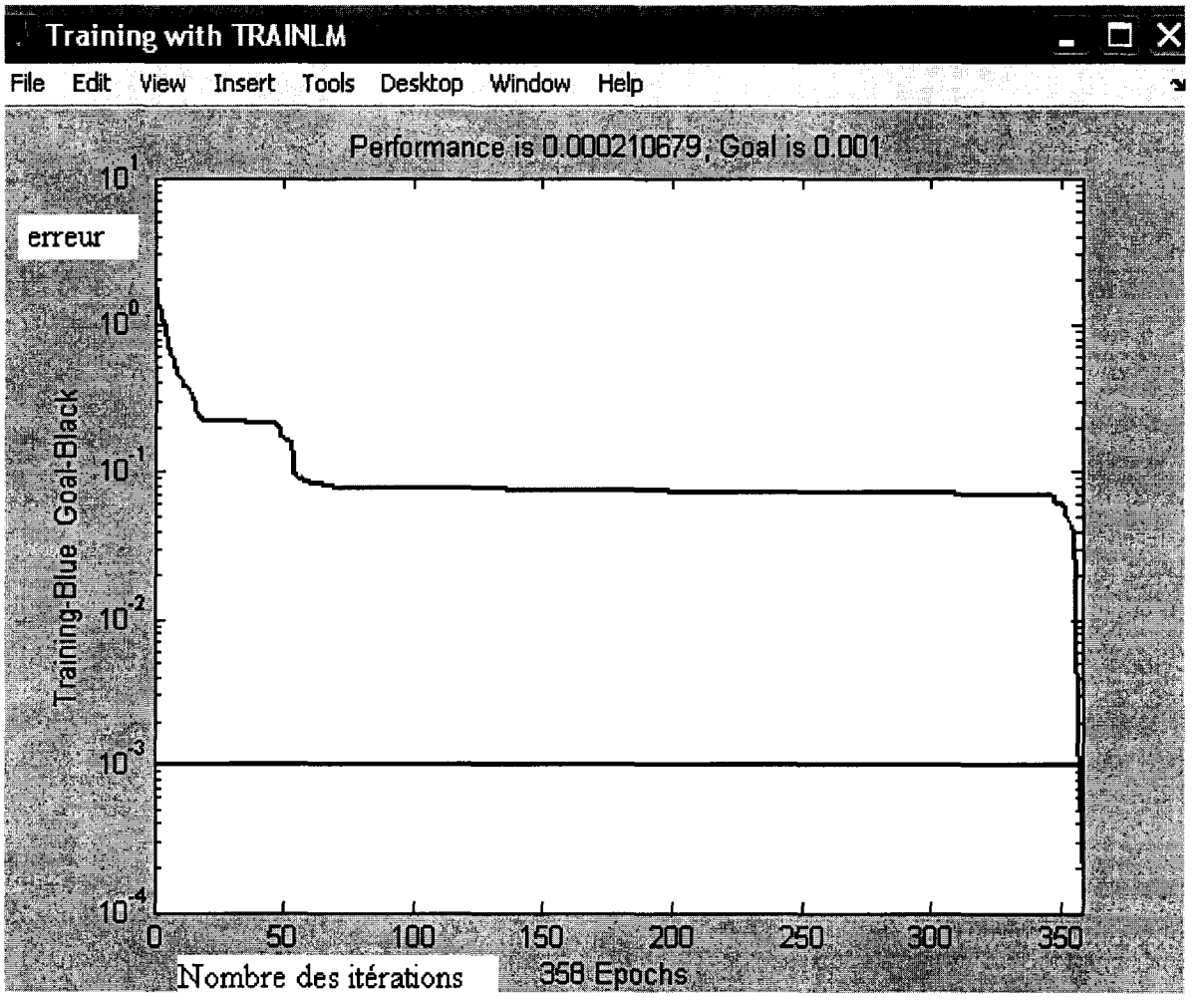



**Figure III. 11 :** Graphe de convergence du modèle supervisé lors de l'apprentissage

Concernant le principe de fonctionnement du réseau, c'est le même principe qui est déjà expliqué précédemment. Dans nos expérimentations la condition d'arrêt était définie par Erreur < 0.001. D'après la figue III.11, on constate que le nombre d'itérations pour satisfaire la condition d'arrêt était de 358.

| *Entrée* / *Sortie* | Valeurs d'entrée et de sortie pour l'apprentissage | | | | |
|---|---|---|---|---|---|
| ***E*** | (9, 10,44) | (8, 10,55) | (9, 10,46) | (9, 10,54) | (9, 10,47) |
| ***S*** | **-0.9703** | **-0.9223** | **-0.9507** | **-0.9555** | **-1.0227** |
| ***E*** | (9, 10,47) | (9, 10,39) | (9, 10,50) | (9, 10,42) | (9, 10,51) |
| ***S*** | **-1.0227** | **-1.0064** | **-0.9903** | **-1.0006** | **-0.9713** |
| ***E*** | (8, 10,51) | (9, 10,41) | (9, 10,60) | (6, 8,47) | (8, 10,60) |
| ***S*** | **-0.9229** | **-0.9901** | **-0.9999** | **1.9124** | **-0.9273** |
| ***E*** | (9, 10,38) | (8, 10,47) | (7, 10,56) | (7, 10,40) | (7, 9,40) |
| ***S*** | **-0.9923** | **-1.0302** | **2.0070** | **2.0085** | **2.0062** |
| ***E*** | (6, 10,46) | (6, 9,37) | (5, 9,40) | (6, 8,40) | (6, 8,55) |
| ***S*** | **1.9972** | **1.9986** | **1.9901** | **1.9968** | **1.9978** |
| ***E*** | (8, 10,46) | (8, 10,45) | (8, 10,40) | (10, 10,56) | (9, 10,40) |
| ***S*** | **-0.9005** | **-1.0142** | **-1.0273** | **-1.0066** | **-1.0120** |
| ***E*** | (9, 10,46) | (9, 10,52) | (8, 10,60) | (9, 10,58) | (1, 5,40) |
| ***S*** | **0.9831** | **-0.9832** | **-0.9689** | **-0.9830** | **0.0829** |
| ***E*** | (5, 8,45) | (9, 10,40) | (9, 10,35) | (8, 9,30) | (7, 9,50) |
| ***S*** | **2.0390** | **-0.9551** | **-0.9301** | **-0.0397** | **2.0097** |

**E** : représente le couple d'entrée
**S** : représente la valeur de sortie

**Tableau III.14 : Quelques valeurs de sortie lors de l'apprentissage du modèle supervisé selon l'algorithme de rétropropagation**

Le tableau III.14 représente quelques valeurs des triplets d'entrée qu'on a pris pour la phase d'apprentissage et les différentes valeurs de sorties. On constate après plusieurs expérimentations que le programme nous a généré trois classes, toutes les sorties qui ont la valeur 0 ou proche d'elle, elles appartient à la classe 1, qui ont la valeur 1 ou proche d'elle, elles appartient à la classe 2, qui ont la valeur -1 ou proche d'elle, elles appartient à la classe 3.

Pour les tests et validations, on va utiliser les 75% des données restantes pour tester la validité de l'algorithme.

| *Entrée* / *Sortie* | Valeurs d'entrée et de sortie pour le test | | | | |
|---|---|---|---|---|---|
| ***E*** | (9, 10,41) | (9, 10,60) | (6, 8,47) | (8, 10,60) | (9, 10,38) |
| ***S*** | **-0.9703** | **-0.9223** | **1.9632** | **-0.9555** | **-1.0227** |
| ***E*** | (8, 10,47) | (9, 10,42) | (5, 9,42) | (9, 10,42) | (10, 10,53) |
| ***S*** | **-0.9933** | **-1.0064** | **2.0031** | **-1.0006** | **-0.9713** |
| ***E*** | (10, 10,60) | (7, 9,42) | (7, 10,46) | (8, 10,48) | (7, 9,40) |
| ***S*** | **-0.9229** | **-0.9901** | **1.9999** | **1.9124** | **1.9273** |
| ***E*** | (7, 10,44) | (9, 10,42) | (6, 10,40) | (7, 10,40) | (6, 9,41) |
| ***S*** | **1.9923** | **-1.0302** | **2.0070** | **2.0085** | **2.0062** |
| ***E*** | (7, 9,42) | (7, 10,46) | (5, 9,40) | (6, 8,40) | (6, 8,55) |
| ***S*** | **1.9972** | **1.9986** | **1.9901** | **1.9968** | **1.9978** |
| ***E*** | (8, 10,46) | (8, 10,45) | (8, 10,40) | (10, 10,56) | (9, 10,40) |
| ***S*** | **-0.9005** | **-1.0142** | **-1.0273** | **-1.0066** | **-1.0120** |
| ***E*** | (9, 10,46) | (9, 10,52) | (8, 10,60) | (9, 10,58) | (1, 5,40) |
| ***S*** | **0.9831** | **-0.9832** | **-0.9689** | **-0.9830** | **0.0829** |
| ***E*** | (9, 10,44) | (7, 10,40) | (6, 10,46) | (5, 7,41) | (8, 8,34) |
| ***S*** | **2.0390** | **1.9551** | **2.0001** | **1.9633** | **2.0097** |

**E** : représente le couple d'entrée
**S** : représente la valeur de sortie

**Tableau III. 15 : Quelques valeurs de tests de validation du modèle supervisé selon l'algorithme de rétropropagation**

La même explication qu'on a déjà présentée précédemment s'applique sur le tableau III.15, pour les différentes valeurs obtenues.

Les quelques individus représentés dans le tableau III.15 sont issus des 75% des données laissées pour le test. Nous constatons que quelque soit les données qu'on a utilisées pour la phase de test et validation, la classification s'est effectuée de façon adéquate. Le programme nous a généré trois classes.

### 3.5. Le modèle non supervisé

Les réseaux dans ce cas s'auto-organisent pour distinguer entre les différentes classes qui constituent le vecteur d'entrée.

L'échantillon de l'apprentissage pour ce modèle est exactement le même que le précèdent cas, pour ce type d'apprentissage la sortie désirée n'est pas nécessaire. Le réseau aura la tâche d'organiser les données d'entrée afin de produire les résultats les plus adéquats.

Pour le principe de création du réseau est le même que le modèle supervisé, il nécessite une série d'expérimentations jusqu'à l'obtention de résultats satisfaisants, après avoir procédé plusieurs tests, le réseau nous a généré trois classes et il est composé de quatre couches tel qu'il est représenté dans la figure III.14.La couche d'entrée est constituée de vingt neurones, la première couche cachée possède quatre neurones, la deuxième couche cachée possède trois neurones, alors que la couche de sortie n'est composée que d'un seul neurone.

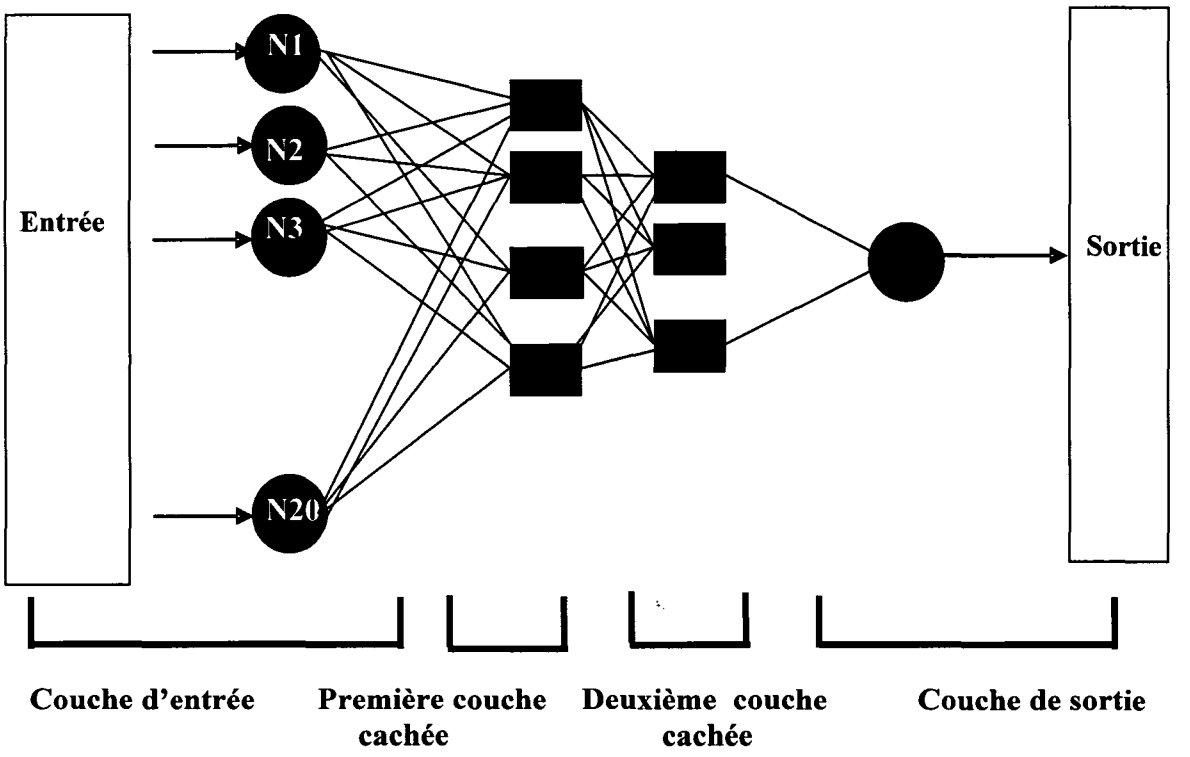



**Figure III.14.** Architecture du réseau pour le modèle non supervisé

Concernant l'apprentissage pour le modèle non supervisé, le nombre d'itérations est 8 pour que l'algorithme converge, comme il est illustré à la figure III.15.

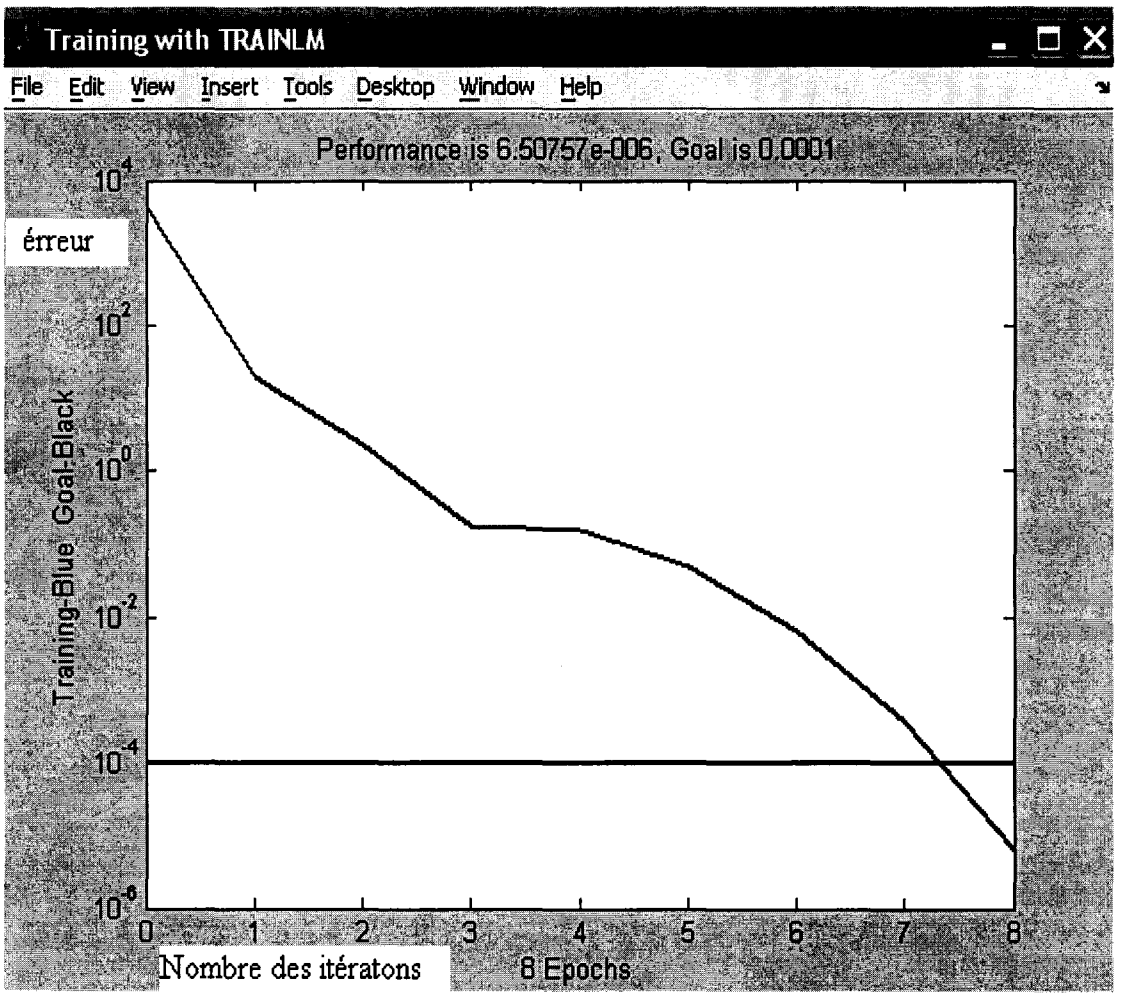



**Figure III. 15 :** Graphe de convergence du modèle non supervisé lors de l'apprentissage

le principe de fonctionnement du réseau, c'est le même principe qui est déjà expliqué précédemment. Dans nos expérimentations la condition d'arrêt était définie par Erreur < 0.001. D'après la figue III.15, on constate que le nombre d'itérations pour satisfaire la condition d'arrêt était de 8.

On peut dire que le modèle non supervisé soit dans la première partie du notre travail ou la deuxième ne nécessite pas beaucoup des itérations et ça revient au principe de fonctionnement du réseau qui s'auto –organise pour aboutir des bons résultats.

| ***Entrée*** / ***Sortie*** | **Valeurs d'entrée et de sortie pour l'apprentissage** | | | | |
|---|---|---|---|---|---|
| ***E*** | (9, 10,44) | (8, 10,55) | (9, 10,46) | (9, 10,54) | (9, 10,47) |
| ***S*** | **0.1123** | **0.1003** | **0.1126** | **0.1131** | **0.1128** |
| ***E*** | (9, 10,47) | (9, 10,39) | (9, 10,50) | (9, 10,42) | (9, 10,51) |
| ***S*** | **0.1128** | **0.1121** | **0.1133** | **0.1123** | **0.1134** |
| ***E*** | (8, 10,51) | (9, 10,41) | (9, 10,60) | (6, 8,47) | (8, 10,60) |
| ***S*** | **0.1002** | **0.1122** | **0.1135** | **0.1801** | **0.1003** |
| ***E*** | (9, 10,38) | (8, 10,47) | (7, 10,56) | (7, 10,40) | (7, 9,40) |
| ***S*** | **01120** | **0.1000** | **0.1855** | **0.1834** | **0.1836** |
| ***E*** | (6, 10,46) | (6, 9,37) | (5, 9,40) | (6, 8,40) | (6, 8,55) |
| ***S*** | **0.1822** | **0.1821** | **0.1800** | **0.1820** | **0.1823** |
| ***E*** | (8, 10,46) | (8, 10,45) | (8, 10,40) | (10, 10,56) | (9, 10,40) |
| ***S*** | **0.1002** | **0.1002** | **0.1001** | **0.1160** | **0.1122** |
| ***E*** | (9, 10,46) | (9, 10,52) | (8, 10,60) | (9, 10,58) | (1, 5,40) |
| ***S*** | **0.1128** | **0.1134** | **0.1003** | **0.1133** | **0.012** |
| ***E*** | (5, 8,45) | (9, 10,40) | (9, 10,35) | (8, 9,30) | (7, 9,50) |
| ***S*** | **0.1801** | **0.1123** | **0.1120** | **0.1001** | **0.1837** |

**E** : représente le couple d'entrée
**S** : représente la valeur de sortie

**Tableau III.16 : Quelques valeurs de sortie lors de l'apprentissage du modèle non supervisé selon l'algorithme de rétropropagation**

Le tableau III.16 représente quelques valeurs des triplets d'entrée qu'on a pris pour la phase d'apprentissage pour le modèle non supervisé et les différentes valeurs de sorties. On constate après plusieurs expérimentations que le programme nous a généré trois classes.

Pour les tests et validations, on va utiliser les 75% des données restantes pour tester la validité de l'algorithme.

| ***Entrée*** / ***Sortie*** | **Valeurs d'entrée et de sortie pour le test** | | | | |
|---|---|---|---|---|---|
| ***E*** | (9, 10,41) | (9, 10,60) | (6, 8,47) | (8, 10,60) | (9, 10,38) |
| ***S*** | **-0.9703** | **-0.9223** | **1.9632** | **-0.9555** | **-1.0227** |
| ***E*** | (8, 10,47) | (9, 10,42) | (5, 9,42) | (9, 10,42) | (10, 10,53) |
| ***S*** | **-0.9933** | **-1.0064** | **2.0031** | **-1.0006** | **-0.9713** |
| ***E*** | (10, 10,60) | (7, 9,42) | (7, 10,46) | (8, 10,48) | (7, 9,40) |
| ***S*** | **-0.9229** | **-0.9901** | **1.9999** | **1.9124** | **1.9273** |
| ***E*** | (7, 10,44) | (9, 10,42) | (6, 10,40) | (7, 10,40) | (6, 9,41) |
| ***S*** | **1.9923** | **-1.0302** | **2.0070** | **2.0085** | **2.0062** |
| ***E*** | (7, 9,42) | (7, 10,46) | (5, 9,40) | (6, 8,40) | (6, 8,55) |
| ***S*** | **1.9972** | **1.9986** | **1.9901** | **1.9968** | **1.9978** |
| ***E*** | (8, 10,46) | (8, 10,45) | (8, 10,40) | (10, 10,56) | (9, 10,40) |
| ***S*** | **-0.9005** | **-1.0142** | **-1.0273** | **-1.0066** | **-1.0120** |
| ***E*** | (9, 10,46) | (9, 10,52) | (8, 10,60) | (9, 10,58) | (1, 5,40) |
| ***S*** | **0.9831** | **-0.9832** | **-0.9689** | **-0.9830** | **0.0829** |
| ***E*** | (9, 10,44) | (7, 10,40) | (6, 10,46) | (5, 7,41) | (8, 8,34) |
| ***S*** | **2.0390** | **1.9551** | **2.0001** | **1.9633** | **2.0097** |

E : représente le couple d'entrée
S : représente la valeur de sortie

**Tableau III. 17 : Quelques valeurs de tests de validation du modèle supervisé selon l'algorithme de rétropropagation**

La même explication qu'on a déjà présentée précédemment tel que présenté dans le tableau III.17, pour les différentes valeurs.

Les quelques individus représentés dans le tableau III.17 sont issus des 75% des données laissées pour le test. On constate que le programme dans ce cas nous a généré trois classes différentes.

## 4. Tache du Data Mining

D'après les résultats obtenues dans le deuxième partie du notre travail, on peut maintenant prendre des différents décisions vis-à-vis l'état de chaque bébé.

Pour les bébés qui appartiennent à la première classe. Ceux qui ont la valeur du *APGAR5* Moins de 4, la décision est la suivante : des manoeuvres lourdes de réanimation sont entreprises et l'enfant sera en l'absence d'amélioration spectaculaire transféré dans un service de réanimation.

Pour les bébés qui appartiennent à la deuxième classe. Ceux qui ont la valeur du *APGAR5* de 4 à 7 , la décision est la suivante : des soins sérieux sont nécessaires et en l'absence rapide d'amélioration l'enfant sera désobstrué, recevra de l'oxygène au masque et sera perfusé.

Pour les bébés qui appartiennent à la troisième classe. Ceux qui ont la valeur du *APGAR5* de 7 à 10, la décision est la suivante :la conduite des médecins sera peu agressive et consistera en une simple désobstruction des voies respiratoires et un apport d'oxygène facultatif.

## 5. Comparaison du notre travail avec la littérature

Généralement, tous les travaux de classification qui sont effectuées soient pour la classification des images, classification textuelle, l'authentification de signatures.....etc., ont suivi le même principe dans leurs travaux; tout d'abord choisir ou préparer les données de telle façon que ces dernières soient bien adaptées avec les différents approches du classification [modèle ART, réseau de neurone, algorithme génitique,...etc.] , après, la création du réseau en spécifiant le nombre de couche qui constitue le réseau et le nombre du neurones pour chaque couche.

Pour notre cas, on est pas loin du l'architecture du réseau utilisé dans la majorité du travaux [10][12][13][14][15], la différence réside dans le nombre du neurone qui constitue chaque couche. La création du l'architecture du réseau dépend généralement de la quantité et la qualité des données lors de la phase de l'apprentissage, l'architecture du réseau pour notre travail est constitué du trois couches avec quinze neurones pour la couche d'entrée, deux neurones pour la couche caché et un seul neurone pour la couche de sortie, pour le modèle supervisé. Dans le cas du modèle non supervisé, l'architecture est la même. La seule différence est dans le nombre de neurones de la couche d'entrée. Cette dernière ne comprend que dix neurones.

Pour la deuxième parties du notre travail, on peut constater les mêmes remarques qu'on a cité précédemment vis-à-vis la première partie.

Concernant les résultats, on a constaté que assez des travaux qui ont atteint un taux de classification de 98% [11] [13], et on a même qui ont atteint un taux de classification presque de 100% [12][14][15] . On peut dire que les résultats qu'on a obtenus sont satisfaisants et encourageante pour élargir notre base de données à un nombre de classe plus grand avec des données plus compliquées.

Nous devons signaler ici l'importance du renforcement des collaborations entre chercheurs des différentes communautés scientifiques concernées par la classification [*Data Mining*], à travers des programmes de recherche pluridisciplinaire tels que le programme Cognisciences du CNRS....etc. Ces recherches pluridisciplinaires devraient favoriser l'apport de solutions nouvelles à des problèmes souvent difficiles, dont les enjeux économiques et sociaux sont considérables.

# CHAPITRE 5

## - Conclusions -

## Conclusions

La classification qu'on a effectuée dans ce travail nous a permis de tirer une conclusion très importante vis-à-vis la matrice de données qu'il faut préparer avant d'effectuer la classification.
La classification des différentes sortes des données, des images, des données médicales, données textuelles...etc. passe par plusieurs étapes, parmi ces étapes, on cite :

- le choix des données;
- la taille des données;
- la préparation des données ou bien la préparation de la matrice finale afin de s'adapter avec l'algorithme choisi pour effectuer la classification.

Notre travail constitue une étape très importante dans le domaine de la classification puisque la majorité des travaux qui effectuent la classification se basent, soit sur le réseau de neurone, modèle ART,...etc. passent par la phase la plus délicate qui est la préparation de la matrice finale qui sera utiliser comme entré pour l'algorithme choisi.
D'après les résultats obtenus, on peut résumer notre conclusion comme suit :

- Les étapes qui précèdent la préparation de la matrice finale se diffèrent d'un chercheur à l'autre,
- La classification des données se diffèrent selon le modèle choisi, soit supervisé ou non supervisé et selon l'algorithme choisi pour effectuer la classification,
- La façon d'interpréter les résultats obtenus se diffère d'un chercheur à l'autre,
- La prise de la décision dépend de la qualité des résultats obtenus et l'objectif à atteindre,
- La préparation de la matrice finale constitue une étape très importante dans la procédure de la classification et les données qui les constituent représentent un grand enjeu pour l'obtention des bons résultats.

Donc, on peut conclu que la qualité des résultats dépend essentiellement de la qualité des données qui constituent la matrice. Les résultats qu'on a obtenu par les deux modèles supervisé et non supervisé prouve cette théorie.

Concernant la première partie du notre travail, nous nous sommes intéressés à l'étude des réseaux de neurones appliqués à la classification et en particulier à la technique de la rétropropagation. Les réseaux de neurone sont reconnus comme l'une des techniques la plus utilisée et la plus efficace parmi les techniques de *Data Mining*. Notre travail consistait à classifier des données mathématiques en utilisant l'algorithme de la rétropropagation avec ces deux variantes, supervisé

et non supervisé. Nous concluons la première partie du notre travail par une étude comparative des deux approches afin de mieux comprendre les subtilités de l'algorithme de rétropropagation.

Les données utilisées pour mener à bien notre étude ont été choisies selon trois critères. Notre premier but était de réduire le plus possible l'impact des données sur la tâche de classification. Des données supposées neutres, nous permettraient de mieux juger de l'efficacité de l'algorithme dans ses deux versions. Notre souci était, aussi, d'éviter la lourdeur de la phase de préparation des données en optant pour des données sans erreurs, sans valeur manquante, sans valeurs nulles et sans redondance. Un troisième point très important pour la classification est d'avoir un volume suffisant de données qui représentent convenablement la population à étudier en quantité et en proportion. La neutralité, la qualité et la quantité proportionnée des données nous ont poussé à choisir des données mathématiques synthétiques spécialement conçues pour la classification.

Notre démarche lors de la construction de l'architecture des deux réseaux était des plus empirique. Elle consistait à choisir de façon temporaire une architecture et d'y apporter des modifications en fonction des résultats obtenus. Cette perpétuelle remise en cause d'une architecture présente, nous a permis après plusieurs itérations de fixer une architecture finale du réseau qui nous a donné les résultats les plus appropriés. Pour le modèle supervisé, les tests répétitifs nous ont mené à un réseau de trois couches. La couche d'entrée est constituée de quinze neurones, la deuxième couche représente la partie cachée et contient deux neurones. La couche de sortie comprend un seul neurone. Dans le cas du modèle non supervisé, l'architecture est sensiblement la même. La seule différence est dans le nombre de neurones de la couche d'entrée. Cette dernière ne comprend que dix neurones au lieu de quinze précédemment.

Pour la phase d'apprentissage, nous avons utilisé, dans les deux cas, un échantillon de 25% de la population de départ. Cette proportion s'est avérée suffisante vu la qualité des données. Les 75% des données restantes nous ont permis de vérifier la véracité de nos différents choix faits précédemment, que ce soit pour les données elles mêmes, l'architecture des réseaux ou la proportion des données d'apprentissage. Les tests effectués nous ont révélé un taux de classification de 100% pour le modèle supervisé et de 91,66% pour le modèle non supervisé.

Les résultats obtenus sont satisfaisants et confirment la réputation des réseaux de neurones en général et celle de l'algorithme de rétropropagation en particulier. Lors de la comparaison entre les deux modèles, nous avons relevés deux différences, l'une majeure et l'autre minime. Pour la première, le taux de convergence de l'algorithme lors de l'apprentissage dans le cas du modèle supervisé est 18 fois plus grand que celui du modèle non supervisé. Cette différence, nous l'avons expliquée par le besoin de l'algorithme dans le cas du modèle supervisé à aller chercher les sorties désirées et prédéfinies par l'utilisateur. Alors que le modèle non supervisé s'auto-organise pour ajuster les sorties afin d'atteindre des résultats appropriés. La deuxième différence est dans le taux de classification, qui reste malgré tout acceptable dans les deux cas. Le modèle supervisé a obtenu un taux de classification parfait de 100% pour toutes les classes, tandis que le taux de classification, dans le modèle non supervisé, a diminué à 83,33 % pour deux des quatre classes.

La deuxième partie du notre travail est consacré à la classification des données médicales afin de prouver notre théorie qu'on a tirée de la première partie.

Notre travail est une nouvelle contribution apportée aux nombreuses études sur la classification et les réseaux de neurones. Pour une évaluation plus complète, il serait judicieux dans le futur d'élargir le volume et le type des données. Comme, il serait pertinent de tester d'autres types de classifieurs, surtout ceux qui supportent les deux modèles supervisé et non supervisé.

## LISTE DES TABLES

## LISTE DES FIGURES

# BIBLIOGRAPHIE


**[1]** Introduction au Data Mining, Michel Jambu, Editions Eyrolles et France Télécom-CENT, 1999, ISBN 2-212-05255-3

**[2]** http://www.si.fr.atosorigin.com/datawarehouse/

**[3]** Le Data Mining, René Lefébure et Gilles Venturi, Editions Eyrolles,Mars 2001,

ISBN : 2-212-09176-1.

**[4]** http://www.eaa.egss.ulg.ac.be/seminaire/docs/Sem03.12.12_pmack.PDF

**[5]** Les réseaux de neurones artificiels et leurs applications en imagerie et en vision par ordinateur, Richard Lepage et Bassel Solaiman, 2003, ISBN 2-921145-40-5.

**[6]** Biskri .I, Delisle .S, " Les n-grams de caractères pour l'extraction de connaissances dans des bases de données textuelles multilingues". TALN 2001, Tours, France.

**[7]** Boné, R., Crucianu, M., Asselin de Beauville, J.-P.: Learning Long-Term Dependencies by the Selective Addition of Time-Delayed Connections to Recurrent Neural Networks. NeuroComputing 48 (2002) 251-266.

**[8]** Lin, T., Horne, B.G., Tino, P., Giles, C.L.: Learning Long-Term Dependencies in NARX Recurrent Neural Networks. IEEE Transactions on Neural Networks 7 (1996) 13-29

**[9]** AntSnio C. Roque-da-Silva-Filho, Use of a Neural Field Model to Derive Equilibrium Values for the Weights of Recurrent Synapses, Universidade de SHo Paulo, Brazil,1990.

**[10]** Juan-Manuel Torres-Moreno, Patricia Veláquez-Morales, « un réseau incrémental pour l'apprentissage non superviséappliqué à la classification de textes », Université du Québec à Montréal ,1999.

[11] Dominic Forest, Jean-Guy Meunier, « La classification mathématique des textes : un outil d'assistance à la lecture et à l'analyse de textes philosophiques», LANCI – Université du Québec à Montréal, 2000.

[12] Sébastien Jouteaul, Antoine Cornuéjols, « Nouveaux résultats en classification à l'aide d'un codage par motifs fréquents», Université Paris-Sud, France, 2003.

[13] Philippe foucher 1,2 , Paul revollon, « Segmentation d'images en couleurs par réseau de neurones :application au domaine végétal», Universitaire de Technologie d'Angers, France, 2000.

[14] Vincent Lemaire, Olivier Bernier «Une nouvelle fonction de coût régularisante dans les réseaux de neurones artificiels », France Télécom Recherche et Développement, France, 2000.

[15] Jean-Luc Bloechle, « réseau de neurones artificiels pour la classification des fontes arabes et la Distinction entre la langue arabe et les Langues latines », Université de Fribourg, Suisse, 2000.

[16] Christophe assens, « le modele du reseau neuronal : un essai de classification des Formes d'auto-organisation reticulaires », Université Paris Dauphine, France, 1994.

[17] Biskri .I, Delisle .S, " Text Classification and Multilinguism: Getting at Words via N-grams of Characters". Proceedings of the 6th World Multi-Conference on Systemic, Cybernetics and Informatics (SCI'02) & the 8th International Conference on Information Systems, Analysis and Synthesis (ISAS'02), Orlando, États-Unis, 2002.

[18] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos, Prabhakar Raghavan, "Automatic subspace clustering of high, dimensional data for data mining applications", Proceedings of ACM SIGMOD, International Conference on Management of Data, 1998.

[19] R. Andrews, J. Diederich, A. Tickle, "A Survey and Critique of Techniques for Extracting Rules from Trained Artificial Neural Networks", Knowledge-Based Systems, 1995.

**[20]** Pavel Berkhin, "Survey of clustering data mining techniques", Technical report, Accrue Software, San Jose, California, 2002.

**[21]** Ming-Syan Chen, Jiawei Han, Philip S. Yu, "Data Mining: An Overview from Database Perspective", Ieee Trans., On Knowledge And Data Engineering, 1997.

**[22]** James Dougherty, Ron Kohavi, Mehran Sahami, "Supervised and Unsupervised Discretization of Continuous Features ", International Conference on Machine Learning,1995.

**[23]** Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, "from Data Mining to Knowledge Discovery: An Overview", Advances in knowledge discovery and data mining, Usama M. Fayyad & al. eds, page 1-34, 1996.

**[24]** Jim Gray, Adam Bosworth, Andrew Layman, Hamid Pirahesh, J. Data Mining and Knowledge Discovery "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals", 1995.

**[25]** Foster Provost, Venkateswarlu Kolluri, "A Survey of Methods for Scaling Up Inductive Algorithms", Data Mining and Knowledge, Volume 3, Issue 2 , page 131-169, Juin 1999.

**[26]** Humberto Luiz Razente, Fabio Jun Takada Chino, Maria Camila Nardini Barioni, Agma J. M. Traina, Caetano Traina Jr., "Visual Analysis of Feature Selection for Data Mining Processes", SBBD , page 33-47, 2004.

**[27]** Celeux .G, Diday .E, Govaert .G, " Classification automatique de données environnement statistique et informatique". Dunod, Informatique, 1989.

**[28]** Celeux .G, Diday .E, Govaert .G, " Classification automatique de données environnement statistique et informatique". Dunod, Informatique, 1989.

[29] Diday .E," Optimisation en classification automatique et reconnaissance de formes". Note Scient. IRIA n° 6, 1972.

[30] Evangelos Simoudis "Reality Check for Data Mining", IEEE Expert: Intelligent Systems and Their Applications, Volume 11, Issue 5, page 26-33, octobre 1996.

[31] Francis Tay, Lixiang Shen, "A Modified Chi2 Algorithm for Discretization" IEEE Transactions on Knowledge and Data Engineering, volume 14, issue 3, page 666-670, Mai 2002.

[32] Ian H. Witten, Eibe Frank "Data mining, practical machine learning tools and techniques with Java implementations", Morgan Kaufmann Publishers, 2000.

[33] Krovetz .R & Croft .W," Lexical ambiguity and information retrieval. Transactions on Information Systems (TOIS)". Publication of the Association for Computing Machinery (ACM), 10 (2), 115–141, 1992.

[34] Lee .M.L, Lu .H, Ling .T.W, Ko .Y.T, " Cleansing Data for Mining and Warehousing". Proceeding of 10th International. Conference on Database and Expert Systems Applications (DEXA), 1999.

[35] Richard Lepage, Bassel Solaiman, " Les Réseaux de Neurones artificiels et Leurs applications en Imagerie et en Vision Par Ordinateur". École de technologie Supérieure , Québec 2003.

[36] Rumelhart .D, Hinton .G, Williams .R, " Learning internal representations by error propagation". In: parallel distributed processing: explorations in the microstructure of cognition. Eds Cambridge, MA: MIT Press, 1986.

[37] Sabrina Tollari, Herve Glotin, Jacques Le Maitre, " Rehaussement de la classification textuelle d'images par leur contenu visuel". Laboratoire SIS - Equipe informatique Université de Toulon, 2004, France.

**[38]** Sahami .M, "Using Machine Learning to Improve Information Access. Ph.d. thesis, Computer Science Department, Stanford University, 1999.

**[39]** Thierry Géraud, Pierre-Yves Strub, Jérôme Darbon, " Segmentation d'Images en Couleur par Classification Morphologique Non Supervisée". International Conference on Image and Signal Processing (ICISP'2001), Agadir, Morocco, May 2001.

**[40]** Humberto Luiz Razente, Fabio Jun Takada Chino, Maria Camila Nardini Barioni, Agma J. M. Traina, Caetano Traina Jr., "Visual Analysis of Feature Selection for Data Mining Processes", SBBD , page 33-47, 2004.

**[41]** Foster Provost, Venkateswarlu Kolluri, "A Survey of Methods for Scaling Up Inductive Algorithms", Data Mining and Knowledge, Volume 3, Issue 2 , page 131-169, Juin 1999.

**[42]** Nicolas Pasquier, "Data mining: Algorithmes d'extraction et de réduction des règles d'association dans les bases de données" Thèse de doctorat, École Doctorale Sciences pour l'Ingénieur de Clermont-Ferrand, Université de Clermont-Ferrand II, Janvier 2000.

**[43]** cf Hébrail & Lechevallier, 2003.

**[44]** Mostafa hanoune, Fouzia benabbou, « modélisation informatique de clients douteux, En utilisant les techniques de datamining », Université Ouejda, 2003..

**[45]** Bruno Agard, Andrew Kusiak , « Exploration des bases de données industrielles à l'aide du data mining – perspectives », École Polytechnique de Montréal & The University of Iowa, 2005.

**[45]** Ismaïl Biskri et Sylvain Delisle, « Un modèle hybride pour le *textual data mining* : un mariage de raison entre le numérique et le linguistique », Université du Québec à Trois-Rivières, juillet 1999.

**[46]**Allman, 1989; Johnson, & Brown, 1988.

**[47]** Grossberg, 1980; Grossberg, 1981.

**[48]** Yves kodratoff, « Extraction de connaissances à partir des données et des textes ("data & text mining") », Université Paris-Sud, France, Mars2002.

**[49]** Jain, & Mao, 1996.

**[50]** Cowan, & Sharp, 1988.

**[51]** Feldman, & Ballard, 1982.

**[52]** Knight, 1990.

**[53]** Alkon, 1989;

**[54]** Ballard, 1984;

**[55]** Fiesler, & Caulfield, 1992;

**[56]** Personnaz, Dreyfus, & Guyon ,1988;

**[57]** Sansonnet, 1988

**[58]** http://sisporto.med.up.pt/