

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Polytechnique

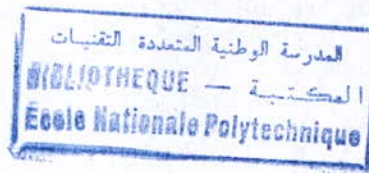
Département : Génie Industriel

P0008

05 B



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique



*Mémoire de fin d'études  
En vue de l'obtention du diplôme  
d'Ingénieur d'Etat.*

*Thème :*

**Conception d'un modèle de prévision basé sur  
les réseaux de neurones artificiels.**

**Application à la consommation des carburants  
(gasoil et essence) à l'horizon 2010.**

**Proposé et Dirigé par :**

M<sup>elle</sup> N. ABOUN

**Etudié par :**

Mr. MILOUDI Adel

Mr. MEGHOUFEL Mâamar

*Promotion 2005*

## *Dédicace :*

*Plusieurs personnes ont su m'encourager, sans faillir, par leur enthousiasme pour ce projet ; je le dédie donc à :*

*Mes chers parents.*

*Mes sœurs.*

*Mon petit frère.*

*Ma famille.*

*Et tous mes amis.*

*Adel.*

## DEDICACE

*A mes chers parents.*

*A mes frères et sœurs.*

*A toute ma famille.*

*A tous ceux que j'aime.*

*Mâamar.*

## REMERCIEMENTS

*Qu'il nous soit permis, au terme de ce mémoire, d'adresser en premier lieu nos sincères remerciements à notre promotrice M<sup>lle</sup> N. ABOUN pour ses conseils judicieux et ses orientations qui nous ont permis de nous aventurer sur des voies profitables.*

*Nous tenons à témoigner notre reconnaissance à Mr. AIB ainsi qu'à Djalel qui nous ont gentiment aidé lors de la collecte des données.*

*Nous tenons également à remercier les membres de nos familles qui nous ont soutenu tout au long de ces années d'études.*

*Enfin, il serait particulièrement ingrat de ne pas remercier tous les professeurs du département Génie Industriel qui ont contribué à notre formation.*

*Adel & Mâamar.*

# Table des matières

**INTRODUCTION GENERALE** ..... 1

## PARTIE THEORIQUE

**CHAPITRE I : Les réseaux de neurones, une nouvelle approche de prévision** ..... 4

- I.1. La prévision ..... 4
- I.2. Les séries chronologiques ..... 4
  - I.2.1. Typologie des chroniques ..... 4
- I.3. Les méthodes classiques de prévision ..... 4
- I.4. Conclusion sur les méthodes classiques de prévision ..... 6
- I.5. Les réseaux de neurones artificiels, une nouvelle approche de prévision ..... 6
  - I.5.1. Principe de base ..... 7
  - I.5.2. Propriété fondamentale ..... 7
    - I.5.2.1. Approximation universelle ..... 8
    - I.5.2.2. Parcimonie ..... 8
- I.6. Fondements théoriques ..... 9
  - I.6.1. Les fonctions d'activation ..... 11
- I.7. Architecture des réseaux de neurones ..... 11
  - I.7.1. Les réseaux de neurones non bouclés ..... 12
  - I.7.2. Les réseaux de neurones bouclés (récurrents) ..... 13
- I.8. Les modes de fonctionnement des réseaux de neurones ..... 14
  - 1.8.1. le mode parallèle ..... 14
  - 1.8.2. le mode séquentiel ..... 14
- I.9. L'apprentissage numérique des réseaux de neurones ..... 14
  - I.9.1. L'apprentissage supervisé ..... 15
  - I.9.2. L'apprentissage non supervisé ..... 16
  - I.9.3. Les règles d'apprentissage ..... 16
- I.10. Conclusion ..... 16

**CHAPITRE II : Les principaux modèles de réseaux de neurones** ..... 17

- II.1. Classification des réseaux de neurones ..... 17
  - II.1.1. Les réseaux FEED-FORWARD ..... 17
    - II.1.1.1. Les Perceptrons ..... 17
      - 1. Le perceptron monocouche ..... 17
      - 2. Le perceptron multicouches (PMC) ..... 17
    - II.1.1.2. Les réseaux à fonction radiale (RBF) ..... 17
  - II.1.2. Les réseaux FEED-BACK ..... 17
    - II.1.2.1. Les cartes auto-organisatrices de Kohonen ..... 17
    - II.1.2.2. Les mémoires associatives de Hopfield ..... 18
- II.2. Entre PMC et RBF ..... 19
  - II.2.1. Les principales différences ..... 19
    - II.2.1.1. Selon leurs fonctions d'activation ..... 19
    - II.2.1.2. Selon la règle d'apprentissage ..... 20
  - II.2.2. Quelques études comparatives ..... 20
  - II.2.3. Le choix final ..... 21

II.3. Le perceptron .....	22
II.3.1. Le perceptron simple .....	22
II.3.2. Le réseau ADALINE .....	24
II.3.3. Perceptron multicouche « PMC » .....	25
II.3.3.1. Le problème du « ou exclusif » .....	25
II.3.3.2. Rétropropagation des erreurs .....	27
II.4. Conclusion .....	29
<b>PARTIE PRATIQUE</b>	
<b>CHAPITRE III : Collecte et traitement des données .....</b>	<b>31</b>
III.1. Introduction .....	31
III.2. Sélection des entrées .....	33
III.2.1. Classement des entrées .....	34
III.2.1.1. Principe de la méthode .....	35
III.2.1.2. Résultats numériques .....	35
III.2.2. Elimination des entrées non pertinentes .....	40
III.2.2.1. principe de la méthode .....	41
III.2.2.2. Résultats numériques .....	41
III.3. Conclusion .....	43
<b>CHAPITRE IV : Conception du modèle neuronal et prévision .....</b>	<b>44</b>
IV.2. La détermination du réseau optimal .....	45
IV.2.1. Choix du nombre de couches cachées du PMC .....	45
IV.2.2.1. Principe de la méthode .....	47
IV.2.2.2. Résultats numériques .....	47
IV.2.3. Sélection du réseau optimal .....	51
IV.3. La prévision de la consommation énergétique .....	56
IV.4. Calcul d'erreurs .....	60
<b>CONCLUSION GENERALE .....</b>	<b>62</b>
<b>ANNEXES</b>	
ANNEXE 1 .....	65
ANNEXE 2 .....	69
ANNEXE 3 .....	73
ANNEXE 4 .....	75
ANNEXE 5 .....	79
ANNEXE 6 .....	81
ANNEXE 7 .....	95
ANNEXE 8 .....	110
<b>REFERENCES BIBLIOGRAPHIQUES.....</b>	<b>114</b>

المدرسة الوطنية المتعددة التقنيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

# *Introduction générale*

## ***Introduction générale :***

Dans la prévision des séries temporelles, il s'agit d'utiliser le passé d'une variable afin d'en extraire des relations permettant de prédire sa valeur future. Cette prévision constitue incontestablement l'appui de toute prise de décision stratégique.

De ce fait, plusieurs méthodes ont été développées pendant les années soixante, bien qu'à cette époque la prévision était plus facile et moins nécessaire, car cette période représente une période à évolution lente où le temps était la meilleure variable explicative d'une croissance appelée à se perpétuer.

En analysant la situation actuelle où l'évolution s'accélère de plus en plus d'une façon loin d'être linéaire et les interdépendances se renforcent, il s'avère nécessaire d'envisager de nouvelles méthodes de prévision.

Pour cela plusieurs recherches se sont orientées vers un autre domaine, plus prometteur, qui est celui des réseaux de neurones formels tirant leur origine de l'intelligence artificielle.

Dans un premier lieu, les réseaux de neurones ont été utilisés pour des tâches de reconnaissance de formes, de paroles et pour la classification. Mais leur capacité à découvrir des formes récurrentes dans les séries, dont la complexité pourrait échapper aux modèles classiques les rend intéressants dans le domaine de l'aide à la décision.

En partant de ce constat, nous avons élaboré ce présent travail dont le but est de trouver jusqu'à quel degré il est possible et utile de remplacer les outils de prévision existants par les réseaux de neurones et ceci dans le cadre de la consommation des carburants (gasoil et essence) dont l'évolution est loin d'être linéaire.

Ainsi, ce document est structuré de la manière suivante :

Une partie théorique, qui comporte deux chapitres :

Le premier chapitre vise à énumérer les différentes méthodes classiques de prévision et leurs limites d'une part, et, d'autre part, à présenter les propriétés fondamentales des réseaux de neurones artificiels, leur apport et leur fondement théorique.

Le deuxième chapitre est consacré aux principaux types de réseaux ainsi qu'au choix et au développement du réseau que nous allons utiliser dans l'application.

Une partie pratique dans laquelle nous avons traité, par une approche neuronale, le problème de la prévision de la consommation des carburants (gasoil et essence) à travers les deux chapitres suivants :



Le troisième chapitre qui comporte la collecte et le traitement des données relatives à l'évolution de la consommation des deux carburants ainsi que les principaux facteurs influençant ce phénomène. Cette partie constitue une étape préalable nécessaire quelque soit l'approche de prévision utilisée.

Dans le dernier chapitre nous exposons la méthodologie adoptée pour la conception de notre réseau neuronal et les résultats obtenus.

Enfin, pour conclure notre document, nous avons présenté l'apport de ce présent travail, ainsi que les limites de l'application des réseaux de neurones artificiels (RNA).

# *Partie théorique*

# CHAPITRE I

## « Les réseaux de neurones, une nouvelle approche de prévision »

### I.1. La prévision :

La prévision vise à déterminer un événement futur à partir du regroupement systématique de données quantitatives. Elle constitue une étape préalable à la planification car elle sert à prévoir ce qui est susceptible d'arriver [GIA 03].

La prévision permet à une entreprise de se définir un éventail d'objectifs réalistes. On désigne aussi par prévision la projection réalisée sur les hypothèses les plus probables en analysant une base de données appelée série chronologique ou temporelle [GUY 90].

### I.2. Les séries chronologiques :

Les séries chronologiques que l'on appelle encore séries temporelles correspondent à une série d'observations effectuées au cours de périodes ou à des instants donnés et qui sont ordonnées selon leurs dates d'observation. Prévoir à partir de chroniques, c'est utiliser les informations du passé d'une ou plusieurs chroniques pour fournir la valeur la plus probable d'une chronique donnée pour une ou plusieurs périodes à venir [GIA 03].

#### I.2.1. Typologie des chroniques : [FIO 99]

Quelque soit la complexité d'une chronique, notée  $X_t$ , on peut toujours distinguer trois composantes principales :

1. *la composante tendancielle* : Elle se décrit par une fonction continue et dérivable dans laquelle le temps est la seule variable explicative.
2. *la composante saisonnière* : Elle est due à des facteurs exogènes apparaissant de façon régulière.
3. *La composante aléatoire  $\varepsilon_t$*  : Sa caractéristique essentielle est d'être non déterministe.

### I.3. Les méthodes classiques de prévision :

Dans le tableau suivant, nous avons classé les différentes méthodes classiques de prévision en présentant les limites de chacune d'elles [GIA 03] [FIO 99] [GUY 90] :

Méthodes	Limites
<p>1. Les méthodes qualitatives :</p> <p>2. Les méthodes quantitatives :</p> <p>    <b>2.1. Méthodes causales :</b></p> <p>    <b>2.2. Méthodes auto projectives :</b></p> <ul style="list-style-type: none"> <li>▪ Moyenne mobile simple :</li> <li>▪ Moyenne mobile double :</li> <li>▪ Lissage exponentiel simple :</li> <li>▪ Lissage exponentiel double :</li> <li>▪ Méthode de Holt &amp; Winters :</li> <li>▪ Méthode de Box &amp; Jenkins :</li> </ul>	<ul style="list-style-type: none"> <li>- Dépendent essentiellement des connaissances des experts d'où la difficulté d'évaluer leur exactitude.</li> <li>- Le grand nombre de paramètres difficiles à mesurer.</li> <li>- Incapable de tenir compte de l'aspect aléatoire des phénomènes économiques.</li> <li>- Applicable uniquement dans le cas d'une chronique horizontale sans tendance ni saisonnalité.</li> <li>- Réagit trop vite aux variations dues au hasard.</li> <li>- Applicable uniquement dans le cas d'une chronique présentant une tendance linéaire.</li> <li>- Donne des résultats imprécis dès qu'une inflexion apparaît dans les données.</li> <li>- Applicable uniquement dans le cas d'une chronique horizontale.</li> <li>- Il n'y a pas de règle précise pour le choix du paramètre.</li> <li>- Applicable uniquement dans le cas d'une chronique présentant une tendance linéaire.</li> <li>- Détermination expérimentale du paramètre.</li> <li>- Choix expérimental des paramètres.</li> <li>- Donne de mauvais résultats en présence d'un comportement tendanciel instable</li> <li>- Les procédures de prétraitement introduisent un biais qu'on ne peut pas toujours définir.</li> <li>- Les prévisions sont construites sur la série transformée et peuvent perdre leur caractère optimal.</li> <li>- La procédure de prétraitement n'est pas toujours possible.</li> <li>- Cette méthode nécessite un historique très consistant avant d'être vraiment opérationnelle et fiable.</li> <li>- Elle est coûteuse et requiert de l'utilisateur une très bonne compréhension des statistiques.</li> </ul>

Tableau I.1 : Les différentes méthodes de prévision et leurs limites.

#### **I.4. Conclusion sur les méthodes classiques de prévision :**

Les méthodes classiques de prévision citées précédemment, bien qu'elles soient toujours utilisées pour des problèmes appropriés, restent limitées dans la qualité de leurs prévisions qui sont loin d'être optimales. Ceci est essentiellement dû :

- A la méthode de redressement préalable de la chronique qui ne suit aucune règle rigide, elle fait partie des compétences de l'utilisateur.
- Au prétraitement de la chronique qui vise à rendre la chronique stationnaire.
- Aux méthodes expérimentales d'estimation des paramètres qui n'assurent pas l'optimalité des paramètres estimés.
- Au choix de la méthode adéquate qui n'est pas toujours évident.

En plus des problèmes de non optimalité, les méthodes classiques souffrent de deux handicaps majeurs. Le premier est que ces méthodes ne peuvent en aucun cas traiter des séries chronologiques non linéaires, c'est-à-dire des chroniques qui ne présentent pas une certaine stabilité d'évolution. En effet, la non stationnarité ainsi que la complexité des processus empêchent le respect rigoureux des hypothèses de base de ces méthodes [COU 99].

Le second handicap est que le résultat ou la prévision donné(e) par ces méthodes ne représente en vérité qu'une vision parcellaire, car les méthodes classiques ne tiennent compte de l'évolution du phénomène étudié qu'à travers la chronique, en ignorant les facteurs qui influencent cette évolution.

#### **I.5. Les réseaux de neurones artificiels, une nouvelle approche de prévision :**

Au début des années quatre-vingt, plusieurs recherches se sont orientées vers le développement de nouvelles techniques basées sur l'approximation de fonctions. Parmi ces techniques figurent les réseaux de neurones dont le domaine d'application ne cesse de s'étendre.

En effet, les réseaux de neurones artificiels (RNA) habituellement utilisés en physique appliquée font leur entrée, depuis une dizaine d'années, dans le domaine économique en tant que méthode quantitative de prévision, à côté des méthodes statistiques classiques [LAD 03].

Il existe plusieurs raisons qui nous ont conduit à nous intéresser à cet outil :

- contrairement aux méthodes statistiques classiques les réseaux de neurones artificiels ne nécessitent aucune hypothèse sur les variables.
- Ils sont tout à fait adaptés pour traiter des problèmes complexes non structurés, c'est-à-dire des problèmes sur lesquels il est impossible a priori de spécifier la forme des

relations entre les variables utilisées. C'est grâce à des algorithmes, que ces systèmes apprennent par eux-mêmes les relations entre les variables à partir d'un ensemble de données, à l'image de ce que ferait le cerveau humain. Ainsi, le réseau se paramètre de lui-même à partir des exemples qui lui sont fournis [DRE 04].

- Il ne répète pas bêtement le passé et s'il existe effectivement une relation de cause à effet entre les descriptions en entrée et les valeurs à prédire, le réseau l'identifiera.
- Le réseau de neurones est robuste. Il n'est pas handicapé par quelques exemples bruités ou faux : ceux-ci sont écartés car ils sont incohérents avec le reste. Les valeurs manquantes sont également gérées habilement et ne nuisent pas à la construction du modèle [DRE 04].
- L'arrêt du processus itératif lorsque le réseau produit les meilleurs résultats sur l'échantillon de validation lui confère une certaine robustesse. On peut en effet considérer que de cette manière, seule l'information pertinente est intégrée dans le système. Les bruits qui pourraient concourir à la spécialisation du réseau ne sont pas intégrés [DRE 04].

### **I.5.1. Principe de base :**

L'idée principale des réseaux de neurones consiste à modéliser une grandeur mesurable (physique, chimique, économique, ...), dont on soupçonne qu'elle dépend d'un ou plusieurs facteurs qui constituent les composantes du vecteur d'entrée.

L'objectif de la modélisation par les réseaux de neurones est de trouver, à partir des mesures disponibles, une relation, si elle existe, entre les variables d'entrée et la grandeur à modéliser. Le cadre conceptuel adapté à cette tâche est celui des statistiques.

En effet, le réseau n'a aucune idée a priori sur le modèle : on choisit une forme d'équation aussi générale que possible, et l'on ajuste les paramètres de cette équation de manière à lui conférer la meilleure capacité de généralisation (estimation des valeurs dans des situations qu'il n'a pas rencontrées) sans s'intéresser particulièrement aux valeurs ou au sens de ces paramètres. On parle alors de modélisation « boîte noire » [DRE 04].

### **I.5.2. Propriété fondamentale :**

Une propriété fondamentale qui distingue un type particulier de réseaux de neurones (celui qui est utilisé pour la prévision) est l'approximation universelle parcimonieuse. Cette expression traduit deux propriétés distinctes : d'une part les réseaux de neurones sont des approximateurs universels, et d'autre part, une approximation à l'aide de réseaux de neurones nécessite, en général, moins de paramètres ajustables que les approximateurs usuels.

### **I.5.2.1. Approximation universelle :**

Toute fonction bornée suffisamment régulière peut être approchée uniformément, avec une précision arbitraire, dans un domaine fini de l'espace de ses variables, par un réseau de neurones avec une certaine architecture [DRE, 04].

Ainsi, on dit que les réseaux de neurones sont des approximateurs universels. Mais, cette propriété n'est pas spécifique aux réseaux de neurones, puisque les polynômes jouissent aussi de cette qualité. Par contre, dans la pratique, on fait toujours en sorte que le nombre de paramètres ajustables soit le plus faible possible. On cherche ainsi, l'approximation la plus parcimonieuse.

### **I.5.2.2. Parcimonie :**

Pour obtenir un modèle non linéaire de précision donnée, un réseau de neurones a besoin de moins de paramètres ajustables. Or le nombre de données nécessaires pour ajuster le modèle est directement lié au nombre de ses paramètres.

En ce sens, le nombre de paramètres, pour une précision donnée, croît exponentiellement avec le nombre de variables dans le cas des approximateurs linéaires par rapport à leurs paramètres (polynômes), alors qu'il croît linéairement avec ce nombre pour les approximateurs non linéaires par rapport à leurs paramètres (réseaux de neurones) [DRE 04].

Mais en quoi cette propriété est-elle avantageuse ?

Au cours de l'apprentissage à partir d'exemples, les paramètres du réseau sont ajustés pour que celui-ci reproduise la relation non linéaire qui existe entre les entrées et les sorties. Or il est clair que le nombre d'exemples nécessaires pour ajuster les paramètres de manière significative doit être grand devant le nombre de ces paramètres. Les réseaux parcimonieux en paramètres sont donc également parcimonieux en nombre d'exemples ; or ceux-ci, très souvent, sont coûteux (notamment si ce sont des résultats d'expériences sur un processus industriel), et leur collecte peut être longue [DRE 04].

L'avantage pratique des réseaux de neurones par rapport aux techniques classiques de modélisation **non linéaire** réside donc dans leur capacité à réaliser des modèles de précision équivalente avec moins de données expérimentales (ou dans leur capacité à constituer des modèles plus précis à partir du même nombre d'exemples). De manière générale, les réseaux de neurones permettent de tirer le meilleur parti des données numériques disponibles, pour construire des modèles [FIO 99].

*En résumé :*

Les réseaux de neurones sont donc un outil puissant d'analyse statistique qui permet d'approximer tout ce qui a un sens et de construire, par la suite, un modèle de prévision à la

fois robuste et flexible.

Avant de mettre à profit cette propriété puissante, nous allons, à travers ce qui va suivre, présenter le principe de fonctionnement de cette nouvelle technique.

## I.6. Fondements théoriques :

Les réseaux de neurones artificiels sont à l'origine une tentative de modélisation mathématique du cerveau humain (cf. *Annexe I*). D'un point de vue technique, il est clair que seuls les principes seront importants. Il ne sera généralement pas nécessaire, pour modéliser telle ou telle fonction, de simuler toutes les molécules chimiques et les enzymes qu'elle implique, l'adéquation fine aux modèles biologiques ne sera retenue que dans la mesure où elle conduit à des réalisations techniquement économiques [LAD 03].

Les premiers travaux datent de 1943 et sont l'œuvre de *Mac Culloch et Pitts*. Ils ont présenté un modèle assez simple pour les neurones et ont proposé la définition suivante :

« Un neurone formel est un automate qui fait une somme pondérée des potentiels d'action qui lui parviennent (chacun de ces potentiels est une valeur numérique qui représente l'état du neurone qui l'a émis), puis s'active suivant la valeur de cette sommation pondérée. Si cette somme dépasse un certain seuil, le neurone est activé et transmet une réponse (sous forme de potentiel d'action) dont la valeur est celle de son activation. Si le neurone n'est pas activé, il ne transmet rien. » [DAV 92].

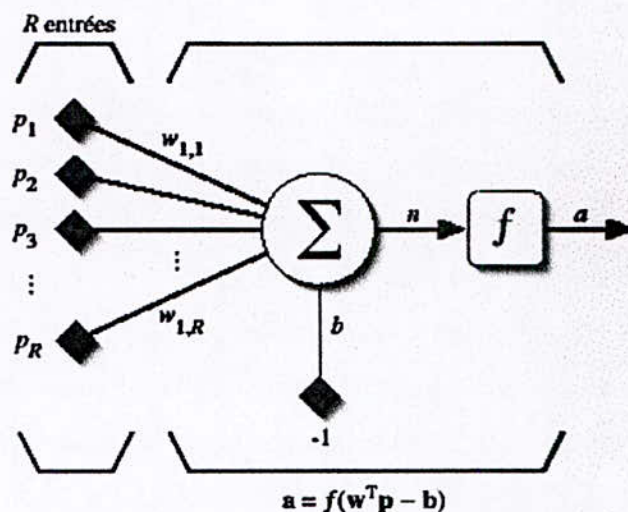


Figure I.1 : modèle du neurone artificiel [HOW 00].

Avec :

- $p_j$  : les entrées du neurone.
- $w_{ij}$  : le paramètre de pondération (le poids de la connexion qui relie le neurone  $i$  à son entrée  $j$ ).



- $b$  : le biais du neurone (seuil d'activation).
- $n$  : le niveau d'activation du neurone.
- $f$  : la fonction d'activation.
- $a$  : la sortie du neurone.

Un neurone est essentiellement constitué d'un intégrateur qui effectue la somme pondérée de ses entrées. Le résultat  $n$  de cette somme est ensuite transformée par une fonction de transfert  $f$  qui produit la sortie  $a$  du neurone. Les  $R$  entrées du neurone correspondent au vecteur  $\mathbf{p} = [p_1 \ p_2 \ \dots \ p_R]^T$ , alors que  $\mathbf{w} = [w_{11} \ w_{12} \ \dots \ w_{1R}]^T$  représente le vecteur des poids du neurone. La sortie  $n$  de l'intégrateur est donnée par l'équation suivante :

$$n = \sum_{j=1}^R w_{1j} p_j - b \quad (\text{I.1})$$

Que l'on peut écrire sous la forme matricielle :

$$n = \mathbf{w}^T \mathbf{p} - b \quad (\text{I.2})$$

Cette sortie correspond à une somme pondérée des poids et des entrées moins le biais  $b$  du neurone. Le résultat  $n$  de la somme pondérée s'appelle le niveau d'activation du neurone. Lorsque le niveau d'activation atteint ou dépasse le seuil  $b$ , alors l'argument de  $f$  devient positif (ou nul). Sinon, il est négatif. Donc la sortie  $a$  du neurone est donnée par :

$$a = f(n) = f(\mathbf{w}^T \mathbf{p} - b) \quad (\text{I.3})$$

L'équation I.3 nous amène à introduire un schéma du modèle (*figure I.2*) plus compact que le premier. On y représente les  $R$  entrées comme un rectangle noir (le nombre d'entrées est indiqué sous le rectangle). De ce rectangle sort le vecteur  $\mathbf{p}$  dont la dimension matricielle est  $R \times 1$ . Ce vecteur est multiplié par une matrice  $\mathbf{W}$  qui contient les poids (synaptiques) du neurone. Le résultat de la multiplication correspond au niveau d'activation qui est ensuite comparé au seuil  $b$  (un scalaire) par soustraction. Finalement, la sortie du neurone est calculée par la fonction d'activation  $f$ . La sortie d'un neurone est toujours un scalaire.

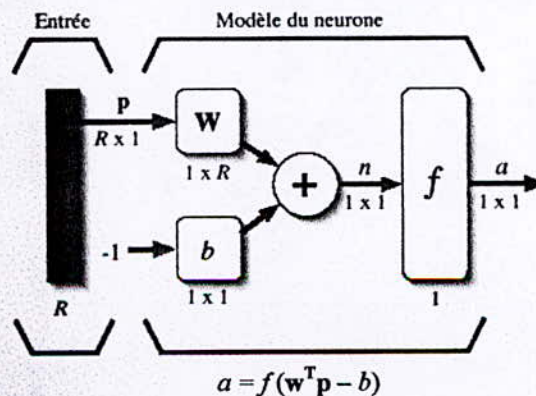


Figure I.2 : Représentation matricielle du modèle d'un neurone artificiel [HOW 00].

### I.6.1. Les fonctions d'activation :

La fonction d'activation définit l'état interne du neurone en fonction de son entrée. Plusieurs fonctions de transfert peuvent être utilisées comme fonction d'activation du neurone, les trois fonctions les plus utilisées sont la fonction à seuil, la fonction linéaire à seuil et la fonction sigmoïde. Le choix de la fonction d'activation dépend de l'application.

En principe toute fonction croissante et impaire peut être utilisée, mais le plus souvent on fait appel à des fonctions ramenant le résultat à l'intérieur de bornes bien définies. Plusieurs fonctions répondant à cet impératif peuvent être envisagées, mais la plus communément utilisée est la fonction sigmoïde (en forme de « s »), symétrique par rapport à 0 sauf si des connaissances spéciales sur le problème conduisent à utiliser une autre fonction, mieux adaptée à celui-ci [DRE 04].

### I.7. Architecture des réseaux de neurones :

Comme nous venons de le voir, un neurone réalise simplement une fonction non linéaire, paramétrée, de ses variables d'entrée. L'intérêt des neurones réside dans les propriétés qui résultent de leur association en réseaux, c'est-à-dire de la composition des fonctions non linéaires réalisées par chacun des neurones [PAR 04].

Un réseau de neurones est donc un maillage de plusieurs neurones et organisé en couches dans la très grande majorité des applications. Pour construire une couche de  $S$  neurones, il s'agit simplement de les assembler comme indiqué dans la *figure 1.3*.

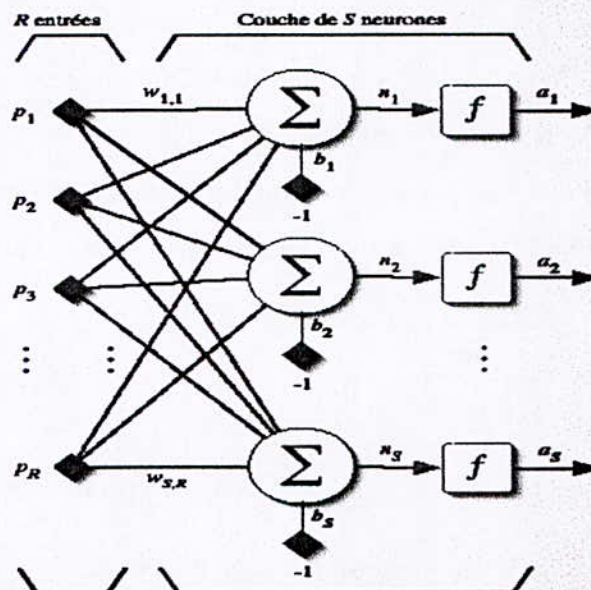


Figure I.3 : Couche de  $S$  neurones [HOW 00].

On distingue deux types de réseaux de neurones : les réseaux non bouclés et les réseaux bouclés :

### I.7.1. Les réseaux de neurones non bouclés :

« Un réseau de neurones non bouclé réalise une (ou plusieurs) fonctions de ses entrées, par composition des fonctions réalisées par chacun de ses neurones. » [DRE 04].

Il est donc représenté graphiquement par un ensemble de neurones connectés entre eux, l'information circule des entrées vers les sorties sans retours en arrière.

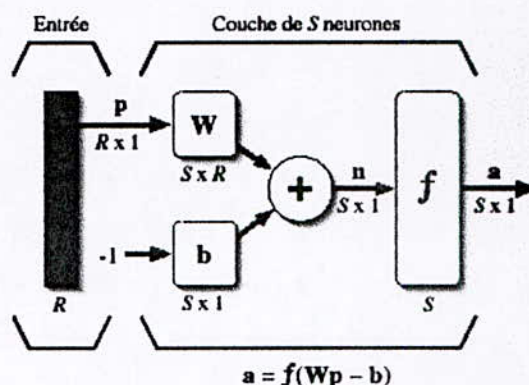
Les  $S$  neurones d'une même couche sont tous branchés aux  $R$  entrées. On dit alors que la couche est totalement connectée. Un poids  $w_{ij}$  est associé à chacune des connexions. Nous noterons toujours le premier indice par  $i$  et le deuxième par  $j$ . Le premier indice (rangée) désigne toujours le numéro de neurone sur la couche, alors que le deuxième indice (colonne) spécifie le numéro de l'entrée. Ainsi,  $w_{i,j}$  désigne le poids de la connexion qui relie le neurone  $i$  à son entrée  $j$ . L'ensemble des poids d'une couche forme donc une matrice  $\mathbf{W}$  de dimension  $S \times R$  :

$$\mathbf{W} = \begin{pmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,R} \\ w_{2,1} & w_{2,2} & \dots & w_{2,R} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ w_{S,1} & w_{S,2} & \dots & w_{S,R} \end{pmatrix}$$

Notons que  $S \neq R$ , dans le cas général (les nombres de neurones et d'entrées sont indépendants). Si l'on considère que les  $S$  neurones forment un vecteur de neurones, alors on peut créer les vecteurs :

$$\mathbf{b} = [b_1, b_2 \dots b_S]^T \quad \mathbf{n} = [n_1, n_2 \dots n_S]^T \quad \mathbf{a} = [a_1, a_2 \dots a_S]^T.$$

Ceci nous amène à la représentation graphique simplifiée, illustrée à la *figure I.4*. On y retrouve, comme à la *figure I.2*, les mêmes vecteurs et matrice. La seule différence se situe au niveau de la taille, ou plus précisément du nombre de rangées ( $S$ ), de  $\mathbf{b}$ ,  $\mathbf{n}$ ,  $\mathbf{a}$  et  $\mathbf{W}$ .



**Figure I.4 : Représentation matricielle d'une couche de S neurones.** [HOW 00]

Finalement, pour construire un réseau, il suffit de combiner plusieurs couches. Chaque couche possède sa propre matrice de poids  $W^k$ , où  $k$  désigne l'indice de couche. Dans le contexte des vecteurs et des matrices relatives à une couche, nous utiliserons toujours un exposant pour désigner cet indice. Ainsi, les vecteurs  $b^k$ ,  $n^k$  et  $a^k$  sont aussi associés à la couche  $k$ .

Il importe de signaler que les couches qui suivent la première ont comme entrée la sortie de la couche précédente. Ainsi, on peut enfileur autant de couches que l'on veut, du moins en théorie. Nous pouvons aussi fixer un nombre quelconque de neurones sur chaque couche. En pratique, comme nous le verrons plus tard, il n'est pas souhaitable d'utiliser trop de neurones. On notera aussi que l'on peut changer de fonction de transfert d'une couche à l'autre [PAR 04].

La couche qui effectue le dernier calcul de la composition de fonctions est nommée «couche de sortie». Les couches qui effectuent les calculs intermédiaires sont nommées «couches cachées».

### I.7.2 Les réseaux de neurones bouclés (récurrents) :

C'est en fait, la forme la plus générale pour un réseau de neurones : lorsqu'on se déplace dans le réseau en suivant le sens des connections, il est possible de trouver au moins un seul chemin qui revient à son point de départ. La sortie d'un neurone du réseau peut donc être fonction d'elle-même : cela n'est évidemment concevable que si la notion de temps est explicitement prise en considération [DRE 04].

**Propriété :**

« Tout réseau de neurones bouclé, aussi complexe soit-il, peut être mis sur une forme canonique, comportant un réseau de neurone non bouclé dont certaines sorties sont ramenées aux entrées par des bouclages de retard unité. » [DRE, 04].

L'élément de base est donc le réseau de neurones non bouclé ; voilà pourquoi l'étude de ce dernier est fondamentale.

**I.8. Les modes de fonctionnement des réseaux de neurones :**

La première hypothèse faite est que le temps est discrétisé et chaque pas de temps est marqué par un top d'horloge. La deuxième hypothèse concerne le mode de fonctionnement, les deux principaux modes sont le mode parallèle et le mode séquentiel [TOU 92].

**1.8.1. le mode parallèle :**

A chaque top d'horloge, tous les neurones calculent leur nouvelle activation, leur sortie et la transmettent aux neurones auxquels ils sont connectés en tant que neurones émetteurs. Ce calcul est fait en fonction de leurs entrées au top précédent.

**1.8.2. le mode séquentiel :**

A chaque top d'horloge un seul neurone calcule sa nouvelle activation, sa sortie, et la transmet aux neurones qui lui sont connectés. Ce calcul étant fait en fonction de ses entrées au top d'horloge précédent.

**I.9. L'apprentissage numérique des réseaux de neurones :**

La notion d'apprentissage est claire et intuitive pour les humains et les animaux : c'est une procédure cognitive qui doit faire en sorte que l'individu réalise, de manière autonome, une tâche donnée. Typiquement, cette procédure s'effectue à partir d'exemples : ainsi, pour apprendre à lire à un enfant, on lui présente des exemples de lettres et de chiffres, écrits avec des écritures et des fentes différentes. A la fin de l'apprentissage, on attend de l'enfant qu'il soit capable de lire non seulement tous les chiffres et lettres de son livre de lecture, mais également tous les chiffres et lettres qu'il est susceptible de voir : en d'autres termes, on attend de lui qu'il ait une capacité de *généralisation* à partir des exemples qui lui ont été présentés, sans qu'il ne soit jamais nécessaire de lui fournir une description analytique et discursive de la forme et de la topologie des chiffres et des lettres [DRE 04].

L'apprentissage numérique poursuit exactement le même objectif : il s'agit, à l'aide d'une procédure numérique programmée et exécutée sur un ordinateur, d'inférer un modèle

d'un processus que l'on observe et sur lequel on peut effectuer des mesures, c'est-à-dire un ensemble d'équations qui décrivent le processus observé et qui permettent de faire des prédictions concernant le comportement de celui-ci.

A cette fin, on fait l'hypothèse que le processus peut être décrit avec la précision désirée par une ou plusieurs fonctions qui contiennent des paramètres, et l'on ajuste ces derniers pour que cette (ou ces) fonction(s) s'ajuste(nt) aux données.

Ainsi, on peut définir l'apprentissage des réseaux de neurones comme étant la procédure qui consiste à estimer (ajuster) les paramètres de ces derniers, afin que celui-ci remplisse au mieux la tâche qui lui est affectée [DRE 04].

Dans le cadre de cette définition, on peut distinguer plusieurs types d'apprentissage, les plus utilisés sont l'apprentissage « supervisé » et l'apprentissage « non supervisé » :

### I.9.1. L'apprentissage supervisé :

L'apprentissage dit supervisé est caractérisé par la présence d'un « professeur » qui possède une connaissance approfondie de l'environnement dans lequel évolue le réseau de neurones. En pratique, les connaissances de ce professeur prennent la forme d'un ensemble de  $Q$  couples de vecteurs d'entrée et de sortie que nous noterons :

$\{(p_1, d_1), (p_2, d_2), \dots, (p_Q, d_Q)\}$ , où  $p_i$  désigne un stimulus (entrée) et  $d_i$  la cible pour ce stimulus, c'est-à-dire les sorties désirées du réseau.

Chaque couple  $(p_i, d_i)$  correspond donc à un cas d'espèce de ce que le réseau devrait produire pour un stimulus donné. Pour cette raison, l'apprentissage supervisé est aussi qualifié d'apprentissage par des exemples.

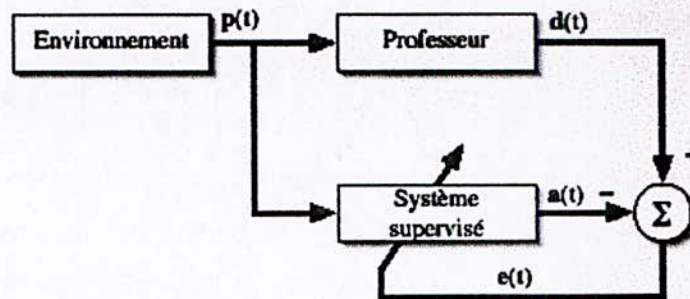


Figure I.5 : Schéma bloc de l'apprentissage supervisé [PAR 04].

L'apprentissage supervisé est illustré d'une manière conceptuelle à la figure I.5. La connaissance de l'environnement par le professeur est graduellement transférée vers le réseau jusqu'à l'atteinte d'un certain critère d'arrêt. Par la suite, on peut éliminer le professeur et laisser le réseau fonctionner de façon autonome [PAR 04].

### **I.9.2. L'apprentissage non supervisé :**

L'apprentissage «non supervisé» ou encore «auto organisé» est caractérisé par l'absence complète de professeur, c'est-à-dire qu'on ne dispose pas d'un signal d'erreur, comme dans le cas supervisé. Nous ne disposons donc que d'un environnement qui fournit des stimuli, et d'un réseau qui doit apprendre sans intervention externe. En assimilant les stimuli de l'environnement à une description de son état interne, la tâche du réseau est alors de modéliser cet état le mieux possible. Pour y arriver, il importe d'abord de définir une mesure de la qualité pour ce modèle, et de s'en servir par la suite pour optimiser les paramètres libres du réseau, c'est-à-dire ses poids synaptiques.

A la fin de l'apprentissage, le réseau a développé une habilité à former des représentations internes des stimuli de l'environnement permettant d'encoder les caractéristiques de ceux-ci et, par conséquent, de créer automatiquement des classes de stimuli similaires [PAR 04].

### **I.9.3. Les règles d'apprentissage :**

Nous venons de voir les principaux types d'apprentissage. Pour chacun de ces types, il existe plusieurs règles qui se distinguent par la manière dont les changements de paramètre surviennent (*cf. annexe 2*). Les règles les plus utilisées sont :

- La règle de HEBB.
- La règle de correction des erreurs.
- La règle d'apprentissage compétitif.

### **I.10. Conclusion :**

Après avoir présenté les éléments de base de l'approche neuronale, dans le chapitre suivant nous allons présenter les principales architectures des réseaux de neurones ainsi que leurs domaines d'application. Ensuite, notre choix portera sur l'architecture la mieux adaptée à des fins de prévision.

# CHAPITRE II

## « Les principaux modèles de réseaux de neurones »

### II.1. Classification des réseaux de neurones :

Comme nous l'avons précisé dans le chapitre précédent, les réseaux de neurones peuvent être classés, selon leurs architectures, en deux grandes familles : les réseaux de neurones non bouclés (FEED-FORWARD), et les réseaux bouclés (FEED-BACK).

#### II.1.1. Les réseaux FEED-FORWARD :

##### II.1.1.1. Les Perceptrons :

##### 1. *Le perceptron monocouche (simple) :*

C'est historiquement le premier RNA, c'est un réseau simple puisqu'il ne se compose que d'une couche d'entrée et d'une couche de sortie.

##### 2. *Le perceptron multicouches PMC :*

C'est une extension du précédent, avec une ou plusieurs couches cachées entre l'entrée et la sortie. Chaque neurone dans une couche est connecté à tous les neurones de la couche précédente et de la couche suivante. Les fonctions d'activation utilisées dans ce type de réseaux sont principalement les fonctions sigmoïdes [LAD 03].

##### II.1.1.2. Les réseaux à fonction radiale :

Ce sont les réseaux que l'on nomme aussi RBF ("Radial Basic Functions"). L'architecture est la même que pour les PMC cependant, les fonctions de base utilisées ici sont généralement des fonctions Gaussiennes. Les RBF seront donc employés dans les mêmes types de problèmes que les PMC [LAD 03].

#### II.1.2. Les réseaux FEED-BACK :

##### II.1.2.1. Les cartes auto-organisatrices de Kohonen :

Ces réseaux ont été introduits pour la première fois par T. Kohonen en 1981. Il s'agit de réseaux non supervisés avec un apprentissage compétitif qui permettent de modéliser l'espace des entrées avec des prototypes et construire une carte à une ou deux dimensions afin de structurer cet espace [PAR 04]. En effet, une carte auto-organisatrice, comme son nom l'indique, s'organise par rapport aux stimuli présentés en respectant les contraintes topologiques de l'espace d'entrée. Il y a mise en correspondance de l'espace d'entrée avec



l'espace du réseau. Les zones voisines de l'espace d'entrée sont voisines sur la carte auto-organisatrice [TOU 92].

### II.1.2.2. Les mémoires associatives de Hopfield :

Les mémoires associatives sont des réseaux récurrents et entièrement connectés. Elles ont été proposées par plusieurs auteurs dès 1977 dont J.J.Hopfield. Dans ce type de réseau, chaque neurone est connecté à chaque autre neurone et il n'y a aucune différenciation entre les neurones d'entrée et de sortie.

Dans mémoire associative, le terme "mémoire" fait référence à la fonction de stockage de l'information et le terme "associative" au mode d'adressage où les motifs mémorisés ne sont pas retrouvés grâce à une adresse, mais directement par le contenu. Ainsi, elles sont capables de trouver un objet stocké en fonction de représentations partielles ou bruitées. L'expression "mémoire adressable par son contenu" (content adressable memory) est aussi employée [COU 97].

Dans le tableau récapitulatif suivant, nous allons présenter les différents réseaux et leurs domaines d'application :

	Types de réseaux	Domaines d'application
<b>Feed-Forward</b>	Perceptron monocouche	– Classification linéaire.
	PMC	– Classification non linéaire. – Approximation de fonction. – Contrôle. – Reconnaissance de formes et de paroles.
	RBF	– Classification non linéaire. – Approximation de fonction. – Contrôle. – Reconnaissance de formes et de paroles.
<b>Feed-Back</b>	Cartes auto-organisatrices	– Projection non linéaire des données. – Compression des données.
	Mémoires associatives	– Problèmes d'optimisation. – Reconnaissance de formes et de paroles.

**Tableau II.1 : Les principaux réseaux et leurs domaines d'application [JAI 96].**

Etant donné que la prévision n'est rien d'autre qu'une approximation de fonctions généralement non connues, les PMC et les RBF qui sont des réseaux à couches et à apprentissage supervisé sont, à partir du tableau II.1, les mieux placés pour le traitement de ce genre de problèmes.

La question qui se pose est : quelles sont les différences entre ces deux architectures et laquelle choisir ?

## II.2. Entre PMC et RBF :

Bien que le PMC et le RBF permettent de résoudre les mêmes types de problèmes, il existe néanmoins quelques différences entre ces derniers. Dans ce qui suit, nous allons essayer de présenter les principaux points de divergence ainsi que quelques études comparatives qui vont guider notre choix du réseau le mieux adapté à la prévision :

### II.2.1. Les principales différences :

#### II.2.1.1. Selon leurs fonctions d'activation :

La principale différence entre ces deux types de réseaux est la manière dont interviennent leurs paramètres [DRE 04] :

- Le PMC est un réseau dont les composantes de base sont des neurones à entrées paramétrées. Pour cette catégorie de neurones, un paramètre (poids) est associé à chaque entrée du neurone. Enfin, une fonction d'activation sigmoïde sera appliquée au potentiel obtenu (somme pondérée des entrées). Pour une fonction d'activation sigmoïde telle que la tangente hyperbolique, nous aurons :

$$a = \text{th} \left( \sum_{i=1}^R w_i p_i \right) \quad (\text{II.1})$$

- Pour le RBF, les paramètres sont attachés à la non linéarité des neurones : ils interviennent directement dans la fonction d'activation. On utilise pour cela des fonctions radiales qui sont généralement des fonctions gaussiennes :

$$a = \exp \left[ - \frac{\sum_{i=1}^R (x_i - w_i)^2}{2\sigma^2} \right] \quad (\text{II.2})$$

La conséquence directe de cette distinction est que les RBF ont des non linéarités locales qui tendent vers zéro dans toutes les directions de l'espace des entrées ; leur zone d'influence est donc limitée dans l'espace, ce qui n'est pas le cas des neurones à fonction d'activation sigmoïde qui possède de ce fait, un meilleur pouvoir de généralisation.

### II.2.1.2. Selon la règle d'apprentissage :

Pour les deux réseaux, l'apprentissage est de type supervisé. Cependant, la différence réside dans la règle d'apprentissage utilisée :

- Pour le PMC, l'apprentissage consiste à estimer les paramètres de la matrice  $W$  en utilisant l'algorithme de la rétropropagation (cf. *Annexe 4*) basé sur la règle de correction des erreurs.
- Pour le RBF, étant donné que deux sortes de paramètres interviennent, l'apprentissage se fait en deux étapes :
  - La première consiste à déterminer les paramètres de la gaussienne (centre et rayons) en calculant la matrice variance-covariance des entrées.
  - La deuxième, quant à elle, calcule les poids de la couche de sortie par correction de l'erreur.

Ainsi, un avantage considérable du RBF par rapport au PMC est que son entraînement est plus simple et plus rapide car il se décompose en deux actions locales combinées ; sauf que cet avantage est dépassé par la puissance des calculateurs actuels. De plus, un inconvénient majeur des réseaux RBF est que, comme les modèles polynomiaux, leur taille croît vite avec la dimension de l'entrée [DEC 03].

### II.2.2. Synthèse de quelques travaux :

Afin de conforter notre choix, nous avons, dans ce qui suit, réuni quelques études comparatives des performances de chacun des deux réseaux dans un domaine d'application particulier :

Une première étude [ROL 96] s'est intéressée à la classification des images multi spectrales par l'approche neuronale. En particulier, une comparaison expérimentale sur les exécutions de plusieurs classificateurs neuronaux est considérée. Les résultats ont montré que le meilleur résultat de classification a été obtenu par le PMC avec une exactitude légèrement meilleur que celle du RBF (une différence de 1%). Or, à travers une autre étude, N.JANKOWSKI et V.KADIRKAMANATHAN [JAN 02] ont montré que pour des problèmes de **classification**, les réseaux RBF peuvent avoir une erreur moindre que celle des PMC dont la performance ne peut dépasser, dans leur application, les 95,4%. De plus, les réseaux RBF même s'ils nécessitent un nombre beaucoup plus important de neurones cachés que les PMC, leur temps de calcul reste relativement court.

Une autre approche [GAW 03] basée sur l'utilisation des RNA pour l'analyse pharmacodynamique d'une population de patients présentant l'échec rénal a été présentée. Deux architectures différentes de réseaux : PMC et RBF ont été étudiés et leurs exécutions

comparées et évaluées statistiquement sur la base de l'erreur de généralisation. L'analyse a indiqué la supériorité du PMC, aussi bien que le suprématie des deux modèles neuronaux sur les modèles linéaires et autorégressifs.

Nous citons aussi, l'étude de M.VERLEYSSEN, E. de BODT et A.LENDASSE intitulée « Prédiction des séries temporelles financières à travers une estimation des dimensions intrinsèques et une projection non linéaire des données » [VER 99] où les auteurs ont utilisé dans un premier lieu un réseau RBF avec lequel ils ont obtenu un taux d'apprentissage de 61% et un taux de 57% pour la généralisation. Ces résultats étaient nettement plus intéressants que ceux qu'ils ont eu avec d'autres méthodes de prédiction, mais les meilleurs résultats restent ceux obtenus par le réseau PMC avec un taux de généralisation supérieur à 62%.

Une dernière étude comparative intéressante et qu'il est utile de citer est celle de P.COULIBALY, F.ANCTIL et B.BOBEE [COU 99]. Cette dernière rassemble plusieurs travaux qui traitent de l'application des RNA dans le domaine de la prédiction et à travers lesquels ils ont tiré diverses conclusions, les plus importantes sont :

- Pour des prévisions à moyen et à long terme, les composantes non linéaires des systèmes et le nombre de variables explicatives prennent plus d'importance, ce qui expliquerait la bonne performance des PMC dans plus de 90% des cas d'études comparatives.
- De nombreuses études expérimentales ont démontré le potentiel des RNA pour la prédiction de certaines séries temporelles. Mais dans 95% des cas d'application en prédiction, seuls des réseaux non bouclés « feed-forward » de type PMC ont été utilisés avec généralement l'algorithme de la rétropropagation.

De plus, les différentes études que nous avons consultées et qui traitent de la prédiction dans le domaine énergétique par les réseaux de neurones optent, pour la plupart d'entre elles, pour l'utilisation des PMC avec l'algorithme de la rétropropagation des erreurs. Nous citons, à titre d'exemple, une étude faite par T.S.DILLON & S.LEUNG [DIL 91] qui utilisent le PMC pour la prédiction de la charge électrique. Dans la littérature, on trouve d'autres études utilisant les mêmes réseaux pour le même type d'application [LEE 91], [SRI 91], [KHE 97] ... Mais le travail le plus important dans ce domaine, reste l'étude japonaise [IWA 99] dans laquelle le PMC apprend le comportement de la charge électrique maximale annuelle de 1975 à 1995 et il est testé sur les trois prochaines années 1996-1998 avant de pouvoir prévoir cette charge à l'horizon 2020.

### **II.2.3. Le choix final :**

Les principales différences entre PMC et RBF ainsi que les études que nous venons d'évoquer, ont dévoilé la supériorité du perceptron multicouches en terme de généralisation et

de précision. En conséquence, nous adopterons ce type de réseaux pour notre application.

Mais avant cette étape, il est nécessaire de présenter, avec un peu plus de détail, la première forme du perceptron, son évolution vers le multicouches et la règle utilisée pour l'apprentissage de ce dernier.

### II.3. Le perceptron :

Le réseau de neurones que nous allons étudier est le «perceptron ». Ce type de réseau appartient la famille générale des réseaux « non bouclé » ou encore «à propagation vers l'avant», c'est-à-dire qu'en mode normal, l'information se propage dans un sens unique, des entrées vers les sorties sans aucune rétroaction. Son apprentissage est de type supervisé par la règle de correction des erreurs.

#### II.3.1. Le perceptron simple :

Le perceptron simple est constitué d'une seule couche de neurones dont les fonctions d'activation sont de type seuils. Il s'agit d'une seule couche de  $S$  neurones totalement connectée sur un vecteur  $\mathbf{p}$  de  $R$  entrées. La matrice  $\mathbf{W}$  de dimension  $S \times R$  représente l'ensemble des poids de la couche. Le vecteur  $\mathbf{b}$  de dimension  $S \times 1$  désigne l'ensemble des  $S$  biais de la couche. Les niveaux d'activation  $\mathbf{n} = \mathbf{W}\mathbf{p} - \mathbf{b} = [n_1 \ n_2 \ \dots \ n_S]^T$  des neurones de la couche servent d'argument à la fonction d'activation qui applique un seuil au niveau 0 pour produire le vecteur des sorties  $\mathbf{a} = [a_1 \ a_2 \ \dots \ a_S]^T$ , où :

$$a_i = \begin{cases} +1 & \text{si } n_i > 0 \\ -1 & \text{autrement} \end{cases} \quad (\text{II.3})$$

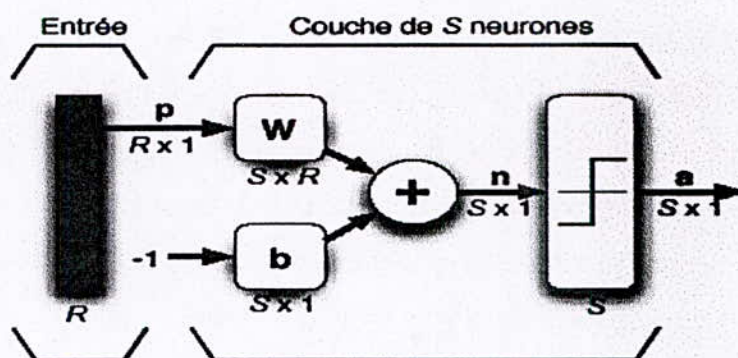
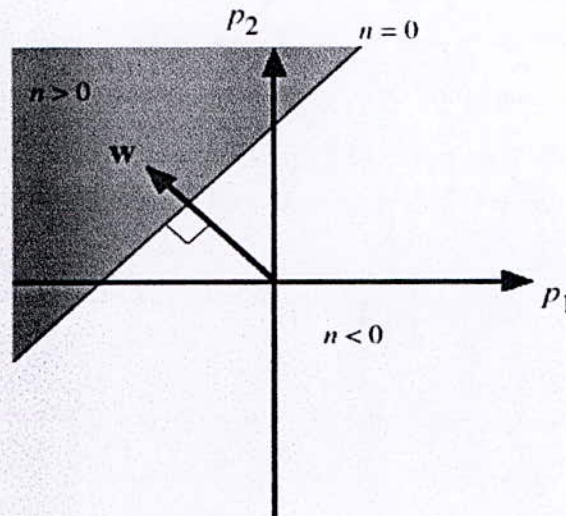


Figure II.1 : Perceptron à une couche avec fonction seuil [HOW 00].

Pour  $R = 2$  et  $S = 1$ , c'est-à-dire lorsque la couche n'est formée que d'un seul neurone relié à deux entrées :  $\mathbf{p} = [p_1, p_2]^T$ ,  $\mathbf{W} = [w_{11}, w_{12}]^T$ ,  $\mathbf{b} = [b_1]$  et  $\mathbf{a} = [a_1]$ , où :

$$a_1 = \begin{cases} +1 & \text{si } w_{11} p_1 + w_{12} p_2 \geq b_1 \\ -1 & \text{autrement.} \end{cases} \quad (\text{II.4})$$

L'équation (II.4) indique clairement que la sortie du réseau (neurone) peut prendre seulement deux valeurs distinctes selon le niveau d'activation du neurone : -1 lorsque ce dernier est strictement inférieur à 0 et +1 dans le cas contraire. Il existe donc dans l'espace des entrées une frontière délimitant deux régions correspondantes. Cette frontière est définie par la condition  $w_{11} p_1 + w_{12} p_2 = b_1$  qui correspond à l'expression générale d'une droite, telle qu'illustrée à la figure II.2 [PAR 04].



**Figure II.2 : Frontière de décision pour un perceptron simple à 1 neurone et 2 entrées** [PAR 04].

En considérant le cas  $S > 1$ , chaque neurone  $i$  possède son propre vecteur de poids et son propre biais, on obtient  $S$  frontières de décision distinctes. Toutes ces frontières de décision sont linéaires et permettent chacune de découper l'espace d'entrée en deux régions infinies, de part et d'autre d'une droite. Chaque neurone d'un perceptron simple permet donc de résoudre parfaitement un problème de classification à deux classes, à condition que celles-ci soient linéairement séparables. Pour cette raison le perceptron simple est souvent appelé « séparateur linéaire ».

Comme nous l'avons dit précédemment, l'apprentissage du perceptron simple est de type supervisé par correction des erreurs. Soit les  $Q$  couples d'apprentissages :

$$\left\{ (p_1, d_1), (p_2, d_2), \dots (p_q, d_q), \dots (p_Q, d_Q) \right\}$$

Pour pouvoir déterminer les poids et les biais du réseau permettant de classer au mieux les  $Q$  couples d'apprentissage, une règle d'apprentissage très simple a été développée. Cette règle consiste à définir un signal d'erreur  $e$ , tel que :

$$e = \frac{d - a}{2}, \quad e \in \{-1, 0, +1\} \quad (\text{II.5})$$

Pour chaque stimulus  $\mathbf{p}_q$  les modifications synaptiques vont se faire de la manière suivante :

$$\Delta \mathbf{w} = \begin{cases} \mathbf{p} & \text{si } e = +1 \\ 0 & \text{si } e = 0 \\ -\mathbf{p} & \text{si } e = -1 \end{cases} \quad (\text{II.6})$$

Dans le cas où  $b \neq 0$ , on peut aussi mettre à jour le biais en observant simplement que celui-ci n'est rien d'autre qu'un poids comme les autres, mais dont l'entrée est fixée à -1 (entrée virtuelle). Ainsi :

$$\Delta b = -e \quad (\text{II.7})$$

Dans le cas général où il y a  $S$  neurones, cette règle peut être réécrite sous forme matricielle de la façon suivante :

$$\begin{aligned} \Delta \mathbf{W} &= \mathbf{e} \mathbf{p}^T \\ \Delta \mathbf{b} &= -\mathbf{e} \end{aligned} \quad (\text{II.8})$$

Où  $\mathbf{e} = \mathbf{d} - \mathbf{a} = [e_1 \ e_2 \ \dots \ e_s]^T$  est le vecteur des erreurs observé en sortie pour le stimulus  $\mathbf{p}$ .

Cette règle converge toujours en un nombre fini d'itérations et permet de trouver un hyperplan (plan dans un espace à  $n$  dimensions) qui sépare ou classifie en deux classes les entrées du perceptron (si ces derniers sont linéairement séparables).

### II.3.2. Le réseau ADALINE :

Nous avons vu que dans le perceptron simple les neurones utilisent une fonction de transfert de type «seuil». En considérant la même architecture de réseau à une seule couche mais avec cette fois-ci une fonction de transfert linéaire. On obtient un réseau qui s'appelle «ADALINE» (en anglais «ADAPtive LInear NEuron»).

Le réseau ADALINE souffre des mêmes limitations que le perceptron simple : il ne peut résoudre que des problèmes linéairement séparables. Cependant, son algorithme d'apprentissage, la règle du «Least Mean Square» (cf. Annexe 3), est beaucoup plus puissante que la règle du perceptron original, car bien que cette dernière soit assurée de converger vers une solution, si celle-ci existe, le réseau résultant est parfois sensible au bruit puisque la frontière de décision se retrouve souvent trop proche des prototypes d'apprentissage

(l'algorithme s'arrête dès que tous les prototypes sont bien classés). En revanche, la règle LMS minimise l'erreur quadratique moyenne, de sorte que la frontière de décision a tendance à se retrouver aussi loin que possible des prototypes [TOU 92].

### II.3.3. Perceptron multicouche « PMC » : [DRE 04]

En traitant les perceptrons à une seule couche de neurones, on a constaté que ces derniers ne pouvaient résoudre que des problèmes de classification linéairement séparables.

Les réseaux multicouches permettent de lever cette limitation. Il est même démontré qu'avec un réseau de trois couches de neurones (02 couches cachées), on peut construire des frontières de décision de complexité quelconque, ouvertes ou fermées, concaves ou convexes, à condition d'employer une fonction de transfert non linéaire et de disposer de suffisamment de neurones sur les couches cachées [PAR 04].

Un réseau multicouche n'est rien d'autre que la mise en cascade de perceptrons simples (à une seule couche), en prenant les sorties d'une couche et en les injectant comme les entrées de la couche suivante.

On peut remarquer qu'il ne sert à rien d'assembler plusieurs couches ADALINE, car la combinaison de plusieurs couches linéaires peut toujours se ramener à une seule couche linéaire équivalente. C'est pourquoi, pour être utile, un réseau multicouche doit toujours posséder des neurones avec fonctions de transfert non linéaires sur ses couches cachées. Sur sa couche de sortie, selon le type d'application, il pourra comporter des neurones linéaires ou non linéaires [PAR 04].

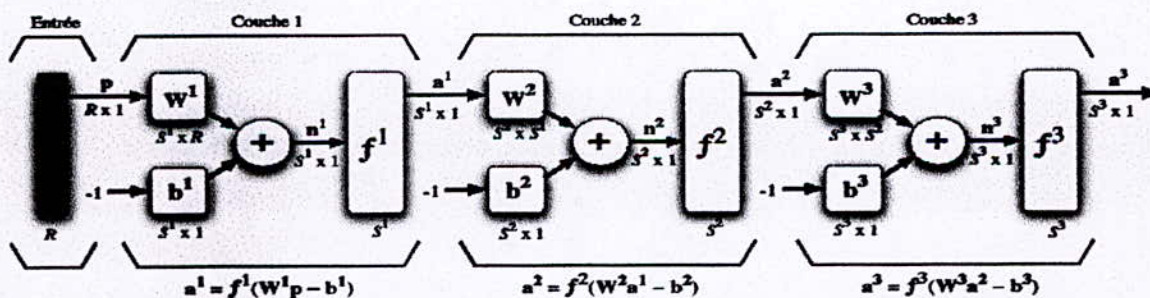


Figure II.3 : Représentation matricielle d'un réseau à trois couches [HOW 00].

#### II.3.3.1. Le problème du « ou exclusif » :

Le premier succès du PMC est la résolution du problème classique du « ou exclusif » (XOR) que l'on ne peut résoudre ni avec un perceptron simple, ni avec un réseau ADALINE. Le « ou exclusif », pour être vrai, exige qu'une seule de ses entrées soit vraie, sinon il est faux. Pour schématiser ce problème on propose que les valeurs 1 (vraie) soient représentées par des points noirs et les valeurs 0 (faux) par des points blancs [TOU 92].



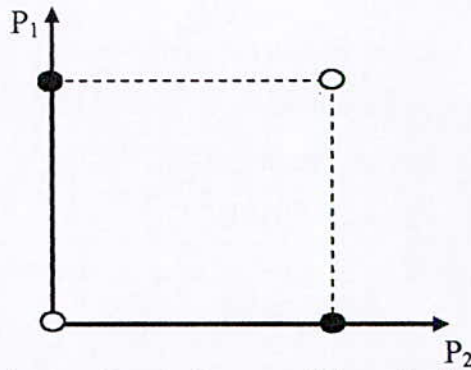


Figure II.4 : représentation graphique du XOR [TOU 92].

Le PMC utilisé pour résoudre ce problème est très simple, il est constitué de trois neurones, dont deux sur la première couche et un seul neurone sur la couche de sortie, son rôle consiste à effectuer la conjonction « et logique » des deux sorties des neurones de la première couche. Ce réseau utilise des fonctions de transfert seuil [PAR 04].

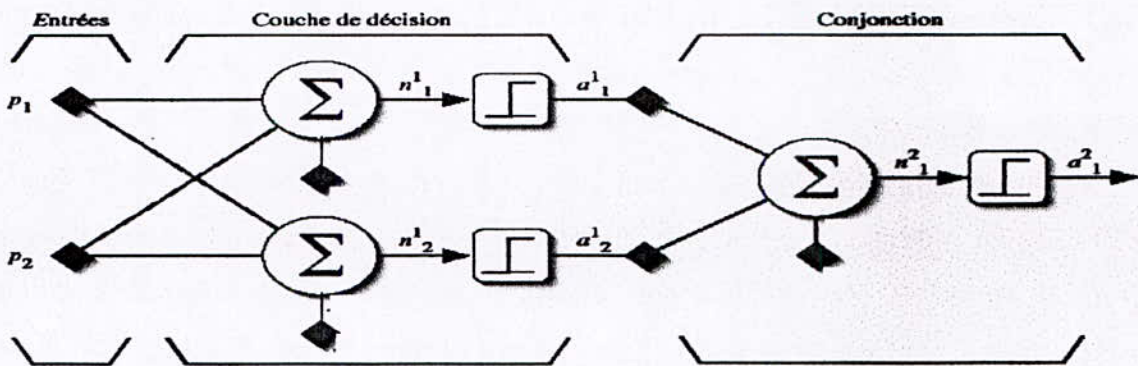


Figure II.5 : PMC pour résoudre le problème du XOR [HOW 00].

Sur la première couche, chaque neurone engendre les frontières de décision illustrées aux figures II.6a et II.6b. Les zones grisées représentent la région de l'espace d'entrée du réseau pour laquelle le neurone correspondant produit une réponse vraie. Le rôle du neurone sur la couche de sortie, illustré à la figure II.6c, consiste à effectuer la conjonction des deux régions produites par les neurones de la première couche [PAR 04].

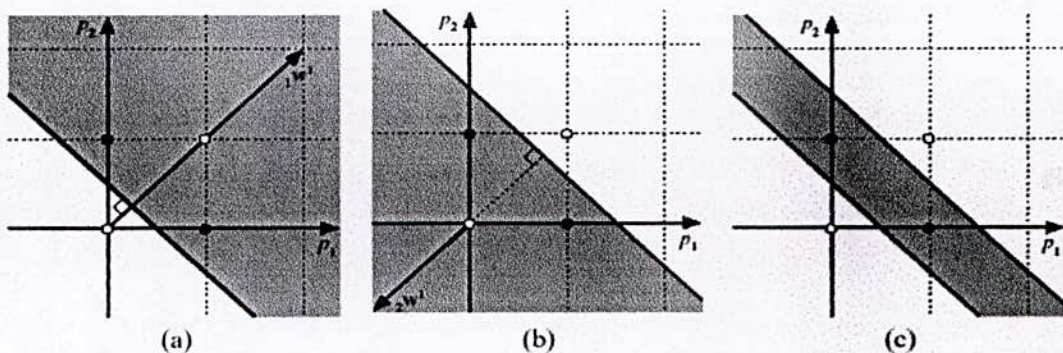
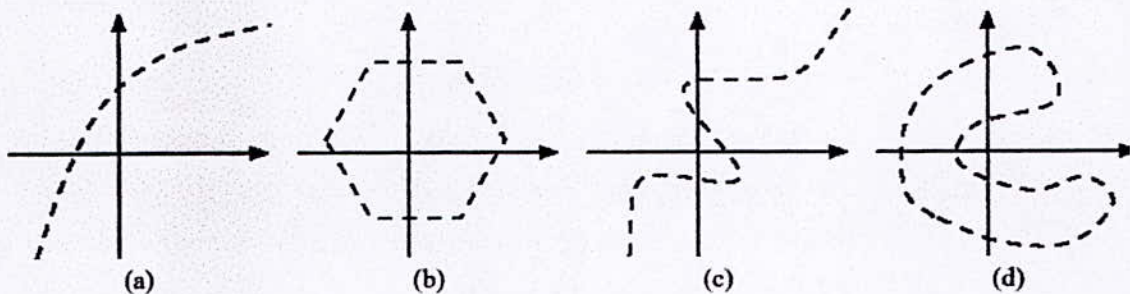


Figure II.6 : frontière de décision engendrée par le réseau de la figure II.5 [PAR 04].

D'une façon générale, pour faire de la classification, on utilise des réseaux soit à deux, soit à trois couches de neurones sigmoïdes. Une seule couche cachée suffit à engendrer des frontières de décision convexes, ouvertes ou fermées, de complexité arbitraire, alors que deux couches cachées permettent de créer des frontières de décision concaves ou convexes, ouvertes ou fermées, de complexité arbitraire. La figure II.7 montre en deux dimensions les différents types de frontières de décision. La première couche cachée d'un tel réseau sert à découper l'espace d'entrée à l'aide de frontières de décision linéaires, comme on l'a vu pour le perceptron simple, la deuxième couche sert à assembler des frontières de décision non linéaires convexes en sélectionnant ou en retranchant des régions engendrées par la couche précédente et, de même, la couche de sortie permet d'assembler des frontières de décision concaves en sélectionnant ou en retranchant des régions convexes engendrées par la couche précédente [HOW 00].



**Figure II.7 : Exemples de frontières de décisions : (a) convexe ouverte ; (b) convexe fermée ; (c) concave ouverte ; (d) concave fermée [PAR 04].**

La puissance de classification du PMC est le résultat direct de sa capacité d'approximation de fonctions. En effet, un réseau multicouche avec une seule couche cachée de neurones sigmoïdes et une couche de sortie avec des neurones linéaires permet d'approximer n'importe quelle fonction avec une précision arbitraire, à condition de disposer de suffisamment de neurones sur la couche cachée [DRE 04].

### II.3.3.2. Rétropropagation des erreurs :

L'algorithme de rétropropagation est une généralisation de la règle LMS. Tous deux utilisent comme indice de performance l'erreur quadratique moyenne, et tous deux permettent un apprentissage de type supervisé avec un ensemble de  $Q$  association stimulus/cible. Mais avant de passer à l'algorithme de rétropropagation pour l'entraînement du PMC, il faut mentionner que ce n'est pas par hasard qu'on remplace la fonction de transfert seuil par la fonction sigmoïde, mais bien pour pouvoir procéder à un apprentissage automatique. Le problème avec la fonction seuil est que sa dérivée est toujours nulle sauf en un point où elle

n'est même pas définie. Elle ne peut donc pas être utilisée avec la méthode de la descente du gradient qui va servir pour développer la méthode de la rétropropagation [BER 01].

La méthode de la rétropropagation des erreurs (cf. Annexe 4) peut être résumée en un algorithme simple qui indique les démarches à suivre pour entraîner un PMC :

1. Initialiser tous les poids du réseau à des valeurs aléatoires (de préférence à de petites valeurs).
2. Pour chaque association ( $\mathbf{p}_q, \mathbf{d}_q$ ) dans la base d'apprentissage :
  - a. Propager les entrées  $\mathbf{p}_q$  vers l'avant à travers les couches du réseau :

$$\begin{aligned} \mathbf{a}^0 &= \mathbf{p}_q \\ \mathbf{a}^k &= f^k(\mathbf{W}^k \mathbf{a}^{k-1} - \mathbf{b}^k) \quad \text{pour } k=1, \dots, M \end{aligned}$$

- b. Rétropropager les sensibilités ( $s_i^k = \frac{\delta \hat{F}}{\delta n_i^k}$ ) vers l'arrière à travers les couches du réseau :

$$\begin{aligned} \mathbf{s}^M &= -2 [\mathbf{f}^M]'(\mathbf{n}^M) (\mathbf{d} - \mathbf{a}^M) \\ \mathbf{s}^k &= [\mathbf{f}^k]'(\mathbf{n}^k) (\mathbf{W}^{k+1})^T \mathbf{s}^{k+1} \quad \text{pour } k = M-1, \dots, 1 \end{aligned}$$

- c. Mettre à jour les poids et biais :

$$\begin{aligned} \Delta \mathbf{W}^k(t) &= -\eta \mathbf{s}^k(t) (\mathbf{a}^{k-1})^T(t) \quad \text{pour } k=1, \dots, M \\ \Delta \mathbf{b}^k(t) &= \eta \mathbf{s}^k(t) \quad \text{pour } k=1, \dots, M \end{aligned}$$

3. Si le critère d'arrêt est vérifié, alors STOP.
4. Sinon, permuter l'ordre de présentation des associations de la base d'apprentissage et aller à l'étape 2 [PAR 04].

Plusieurs critères d'arrêts peuvent être utilisés avec l'algorithme de rétropropagation des erreurs. Le plus commun consiste à fixer un nombre maximum de périodes d'entraînement, ce qui fixe effectivement une limite supérieure sur la durée de l'apprentissage. Un deuxième critère commun consiste à fixer une borne inférieure sur l'erreur quadratique moyenne.

Lorsque l'indice de performance choisi diminue en dessous d'un objectif à atteindre, on considère simplement que le PMC a suffisamment bien appris ses données et on arrête l'apprentissage [PAR 04].

A chaque période d'entraînement, il importe de permuter l'ordre de présentation des stimuli pour réduire la probabilité qu'une séquence de données garde le réseau dans d'un minimum local. En effet, la performance de la méthode de la descente du gradient peut dépendre grandement de cet ordre de présentation qui engendre des trajectoires différentes dans l'espace des paramètres, et des trajectoires différentes peuvent conduire à des minimums

locaux différents. Donc le fait de permuter les données à chaque période nous garantit que l'on ne tombera pas systématiquement sur les mêmes minima. [PAR 04]

#### **II.4. Conclusion :**

Nous venons à travers ce chapitre de justifier le choix du PMC comme étant le réseau le mieux adapté pour des problèmes de prévision. Ensuite, nous avons développé les principes de ce dernier afin de mieux comprendre son fonctionnement. Nous allons, dans ce qui suit, aborder sa mise en œuvre en vue de la prévision de la consommation de la consommation des carburants.

# *Partie pratique*

# CHAPITRE III

## « Collecte et traitement des données »

### III.1. Introduction :

Dans le cadre de notre application, nous nous sommes intéressés à la prévision de la consommation des carburants qui est une des missions essentielles de la SONATRACH. La qualité de cette prévision, élément essentiel de la préparation et de l'anticipation, contribue à garantir l'équilibre production-consommation. Cette prévision est stratégique pour plusieurs raisons :

- La SONATRACH et ses filiales étant les seuls distributeurs.
- Le réseau de transport, moteur de l'économie nationale, est totalement dépendant de cette source d'énergie.
- La décision d'investir sur de nouvelles raffineries ou autres infrastructures pétrolières, dont le coût est énorme, peut dépendre fortement de ces prévisions.
- Elle sert de référence pour vérifier et au besoin adapter le programme de marche de l'ensemble des unités de production afin de satisfaire à l'équilibre offre demande.

Avant de construire notre modèle de prévision, il est impératif d'analyser le comportement passé du phénomène à étudier. Ceci nous permettra de justifier le choix de la méthode de prévision utilisée. Ainsi, nous nous sommes intéressés à l'évolution des carburants les plus consommés, à savoir, le gasoil et l'essence dont les progressions de la consommation de 1980 à 2004, obtenues auprès de la SONATRACH (*cf. Annexe 5*) sont représentées par les deux graphes suivants :

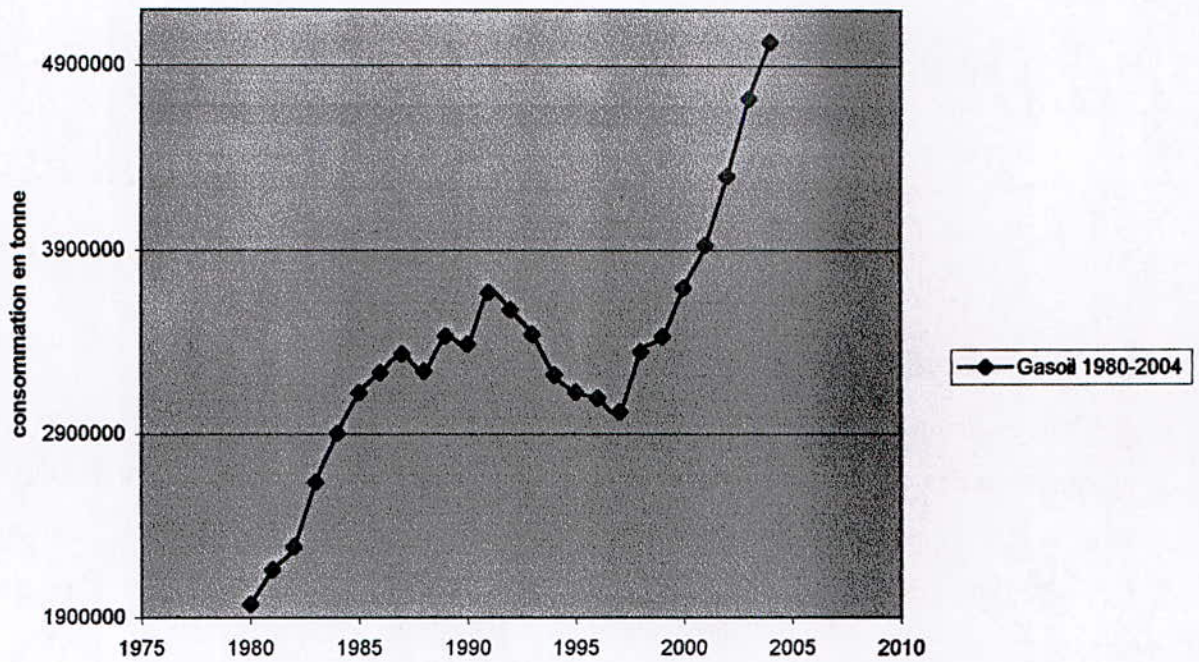


Figure III.1 : Evolution annuelle de la consommation du gasoil

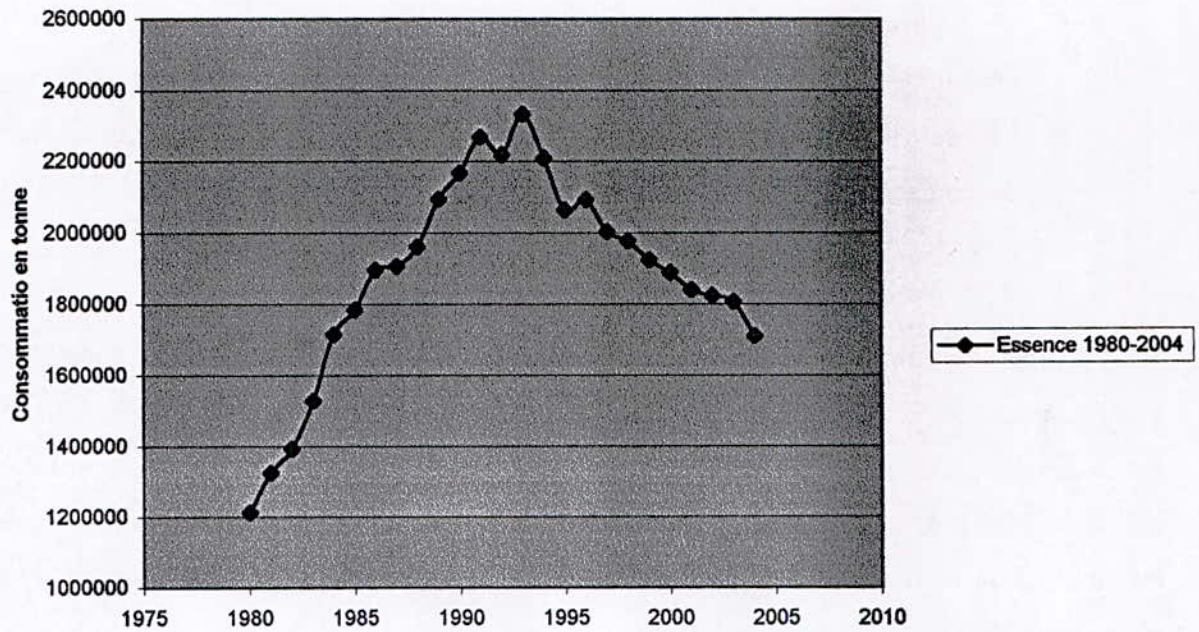


Figure III.2 : Evolution annuelle de la consommation de l'essence

En observant l'évolution de la consommation des carburants (le gasoil et l'essence), nous remarquons que cette dernière a un caractère non linéaire puisque aucune tendance ni

saisonnalité ne peuvent être discernées. Ainsi, une prévision avec une précision satisfaisante ne peut être obtenue par les méthodes classiques.

D'autant plus que la discussion avec des experts dans le domaine énergétique au niveau de la SONATRACH et ceux de l'Agence nationale pour la Promotion et la Rationalisation de l'Utilisation de l'Energie (APRUE) nous a révélé que cette consommation dépend de plusieurs descripteurs de manière non linéaire.

Nous nous sommes aussi rapprochés de l'Office National des Statistiques (ONS), le Centre National de l'Informatique et des Statistiques de la douane (CNIS), de l'APRUE, ... où nous avons pu collecter les données relatives aux évolutions annuelles de 1980 à 2001 des facteurs suivants (*cf. Annexe 5*):

- Population (en millier).
- population active (en millier).
- population urbaine (en millier).
- PIB industriel (en MDA).
- dépense ménages (en MDA).
- dépense du gouvernement à la consommation finale (en MDA).
- parc automobile.

Une bonne méthode de prévision est celle qui peut établir la relation non linéaire entre cette consommation énergétique et les variables exogènes dont elle dépend. L'approche polynomiale dont le nombre de paramètres croît exponentiellement avec le nombre de variables, peut être candidate. Mais étant limités par le nombre d'observations (22 observations) d'une part, et pour plus de précision d'autre part, l'application de l'intelligence artificielle et en particulier les réseaux de neurones artificiels, qui bénéficient de la propriété de parcimonie, s'impose.

Avant d'entamer la procédure de conception d'un modèle neuronal pour chacun des deux carburants, qui sera détaillée dans le prochain chapitre, une phase de sélection des facteurs pertinents s'avère indispensable :

### **III.2. Sélection des entrées :**

Lorsqu'on modélise un processus physique ou chimique, on détermine généralement, par une analyse préalable du problème, les variables qui ont une influence sur le phénomène



étudié : dans ce cas une étape de sélection systématique des entrées n'est pas nécessaire. En revanche ce n'est pas le cas lorsqu'on cherche à modéliser un processus économique [DRE 04].

Pour cela, la stratégie la plus naturelle consiste à partir d'un ensemble de variables candidates aussi grand que possible (on dit alors que l'on considère le « modèle complet ») à comparer les performances de celui-ci à tous les modèles dont les entrées sont des sous-ensembles de l'ensemble des variables candidates et à choisir le meilleur selon un critère bien défini.

Si l'on dispose de  $R$  variables candidates, on peut engendrer  $2^R$  modèles différents, dont il faudrait évaluer et comparer les performances : une telle approche, dont la complexité croît exponentiellement avec le nombre de variables, est certes optimale, mais de mise en œuvre très lourde.

Deux stratégies sous optimales, mais plus simples, sont utilisées en pratique [KAA 95]:

- Une stratégie d'élimination : on cherche à éliminer du modèle complet le descripteur le moins significatif : à cette fin, on compare tous les sous modèles au modèle complet et on conserve le meilleur selon un critère donné.

- Une stratégie de construction : on commence par le modèle le plus simple et on le compare aux  $R$  modèles à un descripteur, on choisit le meilleur, et l'on itère la procédure jusqu'à ce que l'ajout d'un descripteur n'améliore plus la qualité du modèle.

Malgré leur apport, ces deux stratégies restent toujours lourdes à mettre en œuvre.

Dans notre application, nous avons considéré une méthode nécessitant un temps de calcul réduit. Cette dernière, compréhensible de manière intuitive, revêt deux aspects :

- Classement des entrées (facteurs) par ordre d'importance.
- Elimination des facteurs non pertinents.

### III.2.1 Classement des entrées :

Le classement des descripteurs constitue la première étape de la procédure de sélection des entrées choisies. En effet, ce classement nous permet de considérer un nombre très réduit de modèles, et il représente l'intérêt de bien faire prendre conscience de la pertinence relative des descripteurs pour le phénomène à modéliser.

Pour ce faire, plusieurs méthodes peuvent être utilisées en pratique. Dans le cadre de notre travail, nous avons opté pour une méthode efficace et fondée sur des principes solides : *l'algorithme d'orthogonalisation de Gram-Schmidt modifié*. [STU 05]

### III.2.1.1. Principe de la méthode :

Cette méthode est issue des méthodes utilisées pour trouver la solution des moindres carrés d'un problème linéaire par rapport à ses paramètres. C'est une procédure itérative qui classe les descripteurs par ordre décroissant d'importance tout en tenant compte de ceux déjà classés.

Le principe de cette méthode itérative est de choisir à chaque itération, le "meilleur" descripteur puis de supprimer l'influence de ce dernier sur les descripteurs restants.

La mise en œuvre est décrite ci-dessous :

Dans cette méthode, les entrées et la sortie désirée (dont les valeurs sont centrées) sont considérées comme des vecteurs :

$$\mathbf{P} = [p_1, p_2, \dots, p_R] \quad \text{et} \quad \mathbf{d} = [d_1, d_2, \dots, d_Q].$$

*Pour rappel :*

$\mathbf{P}$  : la matrice d'entrée à R descripteurs et Q observations ( $\mathbf{P}$  est donc une matrice  $Q \times R$ ).

$\mathbf{d}$  : le vecteur à Q sorties désirées.

A la première itération, il faut trouver le vecteur d'entrée qui « explique » le mieux la sortie. Pour cela, nous calculons le carré des cosinus des angles entre le vecteur de sortie et les vecteurs d'entrée :

$$\text{Cos}^2(\mathbf{p}_r, \mathbf{d}) = (\mathbf{p}_r^T \mathbf{d})^2 / [(\mathbf{p}_r^T \mathbf{p}_r)(\mathbf{d}^T \mathbf{d})] \quad (\text{III.1})$$

Le vecteur sélectionné est celui pour lequel cette quantité est maximale. Ensuite, nous éliminons la contribution de l'entrée sélectionnée en projetant le vecteur de sortie et tous les vecteurs d'entrée restants sur le sous espace orthogonal au vecteur sélectionné comme suit :

Soit  $\mathbf{p}_r^j$  le vecteur sélectionné à l'itération j, la projection des autres vecteurs d'entrées et celui de la sortie est :

$$\mathbf{p}_q^{j+1} = \mathbf{p}_q^j - [(\mathbf{p}_q^j)^T \mathbf{p}_r^j / (\mathbf{p}_r^j)^T \mathbf{p}_r^j] \mathbf{p}_r^j \quad (\text{III.2})$$

$$\mathbf{d}^{j+1} = \mathbf{d}^j - [(\mathbf{d}^j)^T \mathbf{p}_r^j / (\mathbf{p}_r^j)^T \mathbf{p}_r^j] \mathbf{p}_r^j \quad (\text{III.3})$$

La procédure se poursuit en choisissant, une nouvelle fois, le vecteur d'entrée projeté qui explique le mieux la sortie projetée. Elle se termine lorsque tous les vecteurs d'entrée ont été ordonnés.

### III.2.1.2. Résultats numériques :

A partir de cette base théorique, nous avons élaboré un programme sous MATLAB version 6.5. Le micro-ordinateur utilisé est un pentium IV dont les caractéristiques sont :

- 384 Mo de RAM.
- Processeur intel 2,26 GHZ.

Le programme consiste à réaliser, pour chaque carburant, ce qui suit :

Pour chaque itération :

- Calculer le  $\cos^2$  correspondant à chaque descripteur.
- Ordonner les descripteurs selon la valeur décroissante de leurs  $\cos^2$ .
- Prendre le premier descripteur comme le plus pertinent.
- Projeter le vecteur de sortie ainsi que les descripteurs restants sur le sous-espace orthogonal au descripteur sélectionné et reprendre le calcul avec les nouvelles coordonnées afin de classer les autres descripteurs.

### 1. Développement de l'algorithme de Gram Schmidt pour le gasoil :

#### Itération 1 :

Pour cette première itération, nous avons calculé les valeurs des  $\cos^2$  pour les différents descripteurs à partir du tableau initial des observations, ce qui nous a permis d'établir le premier classement suivant :

Descripteurs	Cos <sup>2</sup>
parc auto	0,7326
essence	0,6609
population	0,5766
Population urbaine	0,5264
Population active	0,4844
PIB industriel	0,2845
dépenses gouvernement	0,2678
dépenses ménages	0,1744

**Tableau III.1 : Classement des descripteurs selon la valeur du  $\cos^2$ .**

Nous remarquons à partir du tableau III.1 que, pour le gasoil, le facteur le plus corrélé avec la sortie est le parc automobile dont le  $\cos^2$  est égal à 0,73. Donc, pour la deuxième itération nous allons projeter les valeurs des descripteurs restants ainsi que celles de la sortie sur le sous espace orthogonal au parc automobile.

## Itération 2 :

années	SORTIE		ENTREES					
	gasoil	essence	population	population active	population urbaine	PIB industriel	dépenses ménages	dépenses gouvernement
1980	-262050	-155170	1580100	1039200	1816500	539600	-6707	162880
1981	-209940	-120850	982400	684700	1200600	382870	-5095	114340
1982	-203880	-121500	576100	435400	721000	248860	-4775	73640
1983	22720	-58510	121900	143900	200900	107720	-3734	29640
1984	138220	37500	-580500	-263500	-565400	-70950	1572	-25690
1985	251550	44470	-852000	-451500	-935200	-193350	4193	-61850
1986	273110	105700	-975600	-584600	-1124600	-297930	10758	-87780
1987	280310	61750	-1167200	-742300	-1332100	-406510	11878	-117950
1988	124330	78630	-1097900	-780800	-1320100	-468930	8675	-135950
1989	240630	168240	-1109100	-874900	-1362600	-526310	7790	-158200
1990	127600	203540	-1097200	-929800	-1367900	-538350	8512	-171660
1991	361350	273120	-911500	-801200	-1181200	-441460	-4948	-155890
1992	227890	201800	-612800	-614600	-881900	-387840	-2097	-126850
1993	56840	294300	-337800	-457300	-616600	-375670	409	-108040
1994	-197960	150330	-33800	-222300	-278900	-273830	-3477	-56690
1995	-331330	-16930	182500	-41500	-24300	-70120	-4182	-4950
1996	-402400	-10470	359100	136000	280400	149160	-2856	67690
1997	-506090	-117510	580500	372900	645500	252150	-2930	97710
1998	-207140	-159240	786000	611400	1011800	222110	-1468	127570
1999	-162710	-236700	907800	818400	1283400	397380	-2973	150120
2000	82960	-281370	1203700	1112600	1694100	858460	-4796	162570
2001	295970	-341130	1495400	1409800	2136700	892940	-3749	225340

Tableau III.2 : Données projetées par rapport au sous-espace orthogonal au parc auto.

A partir du tableau III.2, nous avons calculé les nouvelles valeurs des  $\cos^2$  pour les descripteurs non encore classés :

Descripteurs	$\cos^2$
population	0,3423
dépenses ménages	0,2902
dépenses gouvernement	0,2807
population urbaine	0,2622
population active	0,2223
PIB industriel	0,1560
essence	0,1521

Tableau III.3 : Classement des descripteurs selon le  $\cos^2$ .

Ainsi, le deuxième descripteur est la population dont le  $\cos^2$  égal à 0,34 est maximal.

Pour les autres itérations (de 3 à 7 cf. *Annexe 6*), la même procédure va être suivie jusqu'à aboutir au classement final indiqué dans le tableau suivant :

Pour le Gasoil:	
Classement	Descripteurs
1	Parc auto
2	Population
3	Population urbaine
4	Dépenses gouvernement à la consommation finale.
5	Population active
6	Essence
7	Dépenses ménages
8	PIB industriel

**Tableau III.4 : Classement des descripteurs pour le modèle gasoil.**

2. Développement de l'algorithme de Gram Schmidt pour l'essence :

Itération 1 :

Comme pour le gasoil, dans la première itération, nous avons calculé les valeurs des  $\cos^2$  pour les différents descripteurs qui influencent la consommation de l'essence. Nous avons obtenu le premier classement suivant :

Descripteurs	$\cos^2$
gasoil	0,6609
population	0,4512
Population active	0,2947
Population urbaine	0,3627
PIB industriel	0,0806
dépenses ménages	0,1617
parc auto	0,6601
dépenses gouvernement	0,1008

**Tableau III.5 : Classement des descripteurs selon le  $\cos^2$ .**

Nous remarquons que la consommation de l'essence est influencée par celle du gasoil. C'est d'ailleurs le facteur le plus corrélé avec la sortie, ceci s'exprime par un  $\cos^2$  égal à 0,66.

Donc, pour la deuxième itération nous allons projeter les valeurs des descripteurs restants ainsi que celles de la sortie sur le sous espace orthogonal au gasoil.

Itération 2 :

années	SORTIE	ENTREES						
	Essence 10 <sup>5</sup>	population 10 <sup>6</sup>	population active 10 <sup>6</sup>	population urbaine 10 <sup>6</sup>	PIB industriel 10 <sup>6</sup>	dépenses ménages 10 <sup>4</sup>	parc auto 10 <sup>5</sup>	dépenses gouvernement 10 <sup>5</sup>
1980	-101020	0459200	0412200	0669300	0.2637	-0.2989	-0.6509	0.6126
1981	-081330	-0050300	0116100	0153900	0.1381	-0.1994	-0.6742	0.2492
1982	-075250	-0152300	0018200	-0035500	0.0595	-0.2012	-0.3436	0.0311
1983	-114780	-1578800	-0686100	-1403300	-0.1849	-0.2424	-1.9823	-0.6832
1984	-063360	-2509400	-1172400	-2348300	-0.3691	0.1898	-2.5145	-1.2176
1985	-103850	-3133900	-1501300	-3015800	-0.5196	0.3672	-3.1830	-1.6372
1986	-046030	-3128700	-1564800	-3076200	-0.5951	0.9898	-3.0880	-1.7912
1987	-085800	-3091900	-1608100	-3064900	-0.6613	1.0736	-2.8460	-1.9477
1988	014260	-1914200	-1146500	-2053200	-0.5753	0.8135	-1.2199	-1.6779
1989	051970	-2399200	-1440000	-2506900	-0.6813	0.6481	-2.0324	-2.0263
1990	149290	-1523000	-1102400	-1730000	-0.5750	0.7584	-0.7849	-1.7988
1991	106540	-2569300	-1512400	-2634800	-0.6250	-0.7167	-2.7352	-2.0602
1992	104840	-1376000	-0924200	-1531000	-0.4539	-0.3753	-1.4048	-1.4169
1993	285880	0021100	-0264400	-0258000	-0.2954	-0.0503	0.2724	-0.7912
1994	271690	1923700	0683500	1511700	0.0114	-0.3214	2.6884	0.3309
1995	174190	3040200	1268500	2576000	0.3336	-0.3362	4.0248	1.2044
1996	219190	3744200	1685000	3357300	0.6244	-0.1782	4.7910	2.1490
1997	166260	4661200	2234100	4348000	0.8188	-0.1419	5.8254	2.7237
1998	-022000	3191700	1734900	3224100	0.5835	-0.1517	3.2183	2.4274
1999	-118400	3163200	1880900	3367700	0.7456	-0.3344	2.9426	2.6230
2000	-284010	2064200	1559800	2536400	1.0348	-0.6432	0.7794	2.2477
2001	-448280	1158500	1329500	1913600	0.9224	-0.6494	-1.0825	2.4492

**Tableau III.6 : Données projetées par rapport au sous-espace orthogonal au gasoil.**

A partir du tableau III.6, nous avons calculé les nouvelles valeurs des cos<sup>2</sup> pour les descripteurs non encore classés :

Descripteurs	Cos <sup>2</sup>
parc auto	0,15
PIB industriel	0,0925
dépenses gouvernement	0,0429
population	0,0206
Dépenses ménages	0,014
population active	0,003
population urbaine	0,00095955

**Tableau III.7 : classement des descripteurs selon le Cos<sup>2</sup>.**

Le deuxième descripteur classé est le parc auto avec un cos<sup>2</sup> égal à 0,15.

Pour les autres itérations (de 3 à 7 cf. *Annexe 6*), nous allons procéder de la même façon. Le classement final est indiqué dans le tableau suivant :

Pour l'essence:	
Classement	Descripteurs
1	Gasoil
2	Parc auto
3	Dépenses gouvernement à la consommation finale.
4	Dépenses ménages
5	PIB industriel
6	Population
7	Population active
8	Population urbaine

**Tableau III.8 : Classement des descripteurs pour le modèle essence.**

### III.2.2. Elimination des entrées non pertinentes :

Lorsqu'on conçoit un modèle à partir de mesures, il est indispensable que l'ensemble des variables d'entrée soit aussi réduit que possible : en effet, à chaque entrée du modèle correspond un nombre de paramètres égal au nombre de neurones cachés. Il est donc nécessaire :

- Que la représentation des entrées soit aussi compacte que possible.
- Que l'ensemble des entrées contienne tous les facteurs pertinents par rapport à la grandeur à modéliser, et seulement ceux-ci. En effet, la présence de variables qui ont, sur la grandeur à modéliser, moins d'influence que les perturbations ou le bruit, entraîne celle de paramètres superflus et introduit des variations non significatives sur les variables d'entrée qui constituent autant de sources d'erreur pour le modèle [DRE, 04].

La sélection des entrées pertinentes parmi les R initiales peut être confiée à un test d'hypothèse statistique ou à un autre critère de sélection. Dans cette présente étude nous avons utilisé le test de Fisher, applicable lorsqu'on cherche à comparer un modèle complet aux sous modèles plus restreints. Ces derniers étant issus du modèle complet auquel nous éliminons un par un les entrées les moins pertinentes classées, dans l'étape précédente, par l'algorithme d'orthogonalisation de Gram-Schmidt modifié.

**III.2.2.1. principe de la méthode :**

Pour sélectionner les entrées pertinentes, nous supposons que le modèle complet obéit à l'équation suivante (les vecteurs  $\mathbf{d}$  et  $\mathbf{P}$  étant toujours centrés):

$$\mathbf{d} = \mathbf{P} \boldsymbol{\beta} + \boldsymbol{\omega} \quad (\text{III.4})$$

$\boldsymbol{\beta}$  : vecteur de dimension R des paramètres inconnus du modèle.

$\boldsymbol{\omega}$  : vecteur du bruit.

Tester si l'effet d'une ou plusieurs entrée(s) parmi les R initiale(s) est(sont) statistiquement significatif(s) revient à tester l'hypothèse de nullité des r coefficients correspondants.

On définit :

$\mathbf{d}^{(\text{complet})} = \mathbf{P} \boldsymbol{\beta}^{(\text{complet})}$  : solution des moindres carrés.

$\mathbf{d}^{(\text{incomplet})} = \mathbf{P} \boldsymbol{\beta}^{(\text{incomplet})}$  : solution des moindres carrés sous la contrainte des r coefficients nuls.

Et la variable aléatoire :

$$F_{\text{calculée}} = [(Q-R-1)/r] * [(\|\mathbf{d} - \mathbf{d}^{(\text{incomplet})}\|^2 - \|\mathbf{d} - \mathbf{d}^{(\text{complet})}\|^2) / \|\mathbf{d} - \mathbf{d}^{(\text{complet})}\|^2] \quad (\text{III.5})$$

Pour ce test, les deux hypothèses alternatives sont :

- $H_0$  : les r coefficients sont nuls.
- $H_1$  : les r coefficients ne sont pas nuls.

Si  $H_0$  est vraie, alors la variable aléatoire  $F_{\text{calculée}}$  suit une loi de Fisher Snedecor à r et (Q-R-1) degrés de liberté ; ce qui permet de tester  $H_0$  à partir de la valeur de la réalisation de  $F_{\text{tabulée}}$  dont nous disposons. Si le test conduit à rejeter  $H_0$ , alors le sous modèle est rejeté.

**III.2.2.2. Résultats numériques :**

Nous avons considéré, dans un premier lieu, le modèle complet linéaire par rapport à ses paramètres pour lequel nous allons estimer les paramètres  $\boldsymbol{\beta}^{(\text{complet})}$  par la méthode des moindres carrés qui vont nous permettre de calculer le vecteur  $\mathbf{d}^{(\text{complet})}$ .

A l'itération 1, nous éliminerons du modèle complet le facteur le moins bien classé dans la méthode de Gram Schmidt. Pour ce nouveau modèle nous allons estimer les paramètres  $\boldsymbol{\beta}^{(\text{incomplet})}$  par la méthode des moindres carrés, calculer le vecteur  $\mathbf{d}^{(\text{incomplet})}$  et en dernier lieu, calculer la valeur de «  $F_{\text{calculée}}$  » du modèle qui va être comparée à la valeur «  $F_{\text{tabulée}}$  » correspondante.

Si la condition : «  $F_{\text{calculée}}$  » supérieure à «  $F_{\text{tabulée}}$  » est vérifiée, la procédure s'arrête et le modèle complet sera le modèle optimal. Sinon le nouveau modèle sera considéré



comme étant le modèle complet et nous passerons à la deuxième itération où nous éliminerons le deuxième facteur le moins bien classé et nous retournerons à l'itération 1.

La procédure se poursuit jusqu'à la réalisation du test d'arrêt :

« F calculée » supérieure à « F tabulée »

Le développement numérique du test de Fisher pour les deux modèles gasoil et essence (cf. Annexe 7) nous a permis de trouver les résultats suivants :

*Pour le modèle gasoil :*

Modèle	les sous modèles	F calculée	F tabulée
1	Complet - PIB industriel	0,08858058	244,56
2	Modèle 1 - dépenses ménages	0,14034152	19,42
3	Modèle 2 - essence	1,83173248	8,73
4	Modèle 3 - population active	2,75987193	5,89
5	Modèle 4 - dépenses gouvernement	7,9696905	4,66

**Tableau III.9 : Résultats du F-test pour le modèle gasoil.**

Le modèle final comporte donc, les facteurs suivants :

1. le parc automobile.
2. la population.
3. la population urbaine.
4. les dépenses du gouvernement pour la consommation finale.

*Pour le modèle essence :*

Modèle	les sous modèles	F calculée	F tabulée
1	Complet - population urbaine	0,93185975	244,56
2	Modèle 1 - population active	2,68099389	19,42
3	Modèle 2 - population	2,60637428	8,73
4	Modèle 3 - PIB industriel	2,3562089	5,89
5	Modèle 4 - dépenses ménages	4,12476943	4,66
6	Modèle 5 - dépenses gouvernement	45,5038785	3,98

**Tableau III.10 : Résultats du F-test pour le modèle essence.**

Le modèle final comporte donc, les facteurs suivants :

1. le gasoil.
2. le parc automobile.
3. les dépenses du gouvernement pour la consommation finale.

### **III.3. Conclusion :**

La sélection des entrées constitue la première étape de toute conception de modèle, une fois cette sélection effectuée, nous pouvons passer à la conception du réseau neuronal que nous allons utiliser, par la suite, à des fins de prévision.

# CHAPITRE IV

## « Conception du modèle neuronal et prévision »

Dans ce dernier chapitre, nous allons décrire la méthodologie adoptée afin d'aboutir au réseau optimal relatif à chaque carburant.

Les principales étapes de notre démarche se résument à :

1. La description du type d'apprentissage de l'architecture choisie (PMC) ainsi que le programme MATLAB élaboré à cette fin.
2. La détermination du réseau optimal, cette étape comprend :
  - Le choix du nombre de couches cachées.
  - Le choix du nombre de neurones dans chaque couche (on dit qu'on choisit une famille de réseaux).
  - La sélection du réseau optimal parmi ceux de la famille choisie.
3. La prévision.
4. Le calcul d'erreurs.

### IV.1. Le type d'apprentissage :

Comme nous l'avons vu dans la première partie de ce document, estimer les paramètres d'un modèle : c'est trouver des valeurs numériques pour les composantes de la matrice des paramètres  $W$ , telles que ce modèle soit satisfaisant (selon un critère de satisfaction donnée). A cet effet, on minimise la fonction de coût représentative des différences entre les valeurs mesurées et les valeurs calculées par le modèle. Puisque notre choix s'est porté sur le perceptron multicouches PMC (*cf. Chapitre II*), nous utiliserons, par conséquent, la règle de rétropropagation des erreurs.

Pour cela, nous disposons d'un ensemble de 22 mesures de la sortie ( $d_i$ ,  $i = 1$  à 22) correspondantes à 22 valeurs des entrées ( $p_i$ ,  $i = 1$  à 22), ces mesures représentent les données annuelles relatives à l'évolution de chaque facteur d'entrée de 1980 à 2001.

Les mesures étant entachées de bruit, il n'est pas souhaitable de chercher le modèle dans une famille de réseaux suffisamment complexe pour que le minimum de la fonction de coût soit nul : un tel modèle reproduirait le bruit et les perturbations, alors que l'objectif de

la modélisation est bien au contraire, d'extraire tout ce qui est déterministe dans l'information présente dans les données.

Ce qu'il faut noter aussi, c'est qu'avant tout apprentissage, il est indispensable de normaliser et de centrer toutes les variables d'entrée. En effet, si les entrées ont des grandeurs très différentes, celles qui sont « petites » n'ont pas d'influence sur l'apprentissage. En pratique, il est donc recommandé, pour chaque vecteur d'entrée  $\zeta_i$  (un vecteur d'entrée  $\zeta_i$  comprend toutes les valeurs mesurées de l'entrée correspondante  $i$ ), de calculer la moyenne  $\mu_i$  et l'écart type  $\sigma_i$  de ces composantes, et d'effectuer le changement de variable  $\zeta_i' = (\zeta_i - \mu_i) / \sigma_i$  (ou tout autre changement de variable ayant un effet analogue). De plus, cette normalisation nous permet d'éviter le phénomène de saturation des neurones dans lequel ces derniers cessent d'apprendre tellement leur convergence devient lente.

Le programme MATLAB, que nous avons élaboré pour l'apprentissage, permet de faire apprendre au réseau l'évolution de la consommation du carburant en fonction de l'évolution des entrées du réseau.

Pour l'arrêt de la procédure d'apprentissage, nous avons fixé les critères suivants :

- Un nombre maximal d'itérations (ou époques) égal à 150 itérations.
- Une limite inférieure de la variation du module du gradient égale à  $10^{-10}$  (l'algorithme n'évolue plus).
- Une limite inférieure de la variation du module de l'erreur quadratique moyenne fixé par défaut par MATLAB.

## IV.2. La détermination du réseau optimal :

### IV.2.1. Choix du nombre de couches cachées du PMC :

Pour le choix du nombre de couches cachées, nous nous sommes basés sur le théorème suivant :

« Toute fonction bornée suffisamment régulière peut être approchée uniformément, avec une précision arbitraire, dans un domaine fini de l'espace de ses variables, par un réseau de neurones comportant une couche de neurones cachés en nombre fini, possédant tous la même fonction d'activation, et un neurone de sortie linéaire. » [DRE 04].

Cette propriété montre qu'une seule couche cachée est largement suffisante pour résoudre des problèmes d'approximation de fonctions et donc de prévision.

De plus, on montre qu'on peut améliorer un peu les performances en augmentant le nombre de couches cachées, tout en diminuant le nombre de neurones par couche. Cette

procédure apporte parfois une amélioration, généralement marginale. En revanche, si l'on n'a pas obtenu des résultats satisfaisants avec une couche cachée, il est illusoire d'espérer en obtenir en augmentant le nombre de couches cachées [DRE 04].

Donc, dans notre application, nous allons, à juste titre, modéliser le phénomène de la consommation de chaque carburant par un perceptron à une seule couche cachée.

#### IV.2.2. Choix de la famille de réseaux :

A présent, nous allons aborder le problème du dimensionnement des deux réseaux, c'est-à-dire, celui de la détermination du nombre de neurones dans chaque couche.

Pour chaque année  $t$  et pour chaque type de carburant, les entrées correspondent aux valeurs, pour cette même année, des descripteurs retenus par le test de Fisher ainsi que la valeur de la consommation pendant l'année  $t-1$ . Ceci nous donne :

- 5 entrées pour le gasoil.
- 4 entrées pour l'essence.

La sortie du réseau est, bien entendu, la valeur de la consommation du carburant pendant l'année  $t$ .

Ayant déterminé le nombre d'entrées et de sortie des deux réseaux, nous passons à présent au problème de la détermination du nombre de neurones dans la couche cachée.

La démarche la plus raisonnable consiste à chercher d'abord l'approximation dans une famille de réseaux simples, ensuite, si le résultat obtenu n'est pas satisfaisant, on accroît progressivement la complexité (augmentation du nombre de neurones dans la couche cachée).

Pour notre part, nous allons procéder de la manière suivante :

1/ Afin de minimiser le nombre de réseaux pour lesquels nous devons effectuer l'apprentissage, nous allons déterminer l'intervalle dans lequel se trouve le nombre optimal de neurones cachés. Cet intervalle est déterminé par la règle de la pyramide géométrique [KAA 95], qui se base sur le nombre d'entrées et de sorties du réseau :

$$S = \sqrt{\text{entrées} * \text{sorties}}$$

$$\text{Borne inf} = S/2$$

$$\text{Borne sup} = S*2$$

$$\text{Nombre de neurones cachés} \in [S/2, S*2]$$

2/ Nous choisissons ensuite, une structure de réseau (nombre de neurones cachés) et nous estimons les paramètres de ce dernier. Nous procédons ainsi, en partant de la structure la plus simple ( $S/2$  neurones cachés) et en rendant la structure progressivement plus complexe ( $S*2$  neurones cachés). Pour chaque structure, nous choisissons le meilleur

réseau. Ainsi, nous pouvons comparer les structures selon leur meilleur réseau.

Pour ce faire, il existe une approche qui permet la sélection d'un réseau parmi plusieurs : « *la validation croisée* ».

#### IV.2.2.1. Principe de la méthode :

Au lieu de fonder la sélection du réseau sur l'erreur commise sur les données d'apprentissage, la validation croisée est une méthode d'estimation de l'erreur de généralisation à partir de données qui ne sont pas utilisées pour l'estimation des paramètres.

En effet, l'erreur d'apprentissage peut être rendue arbitrairement petite en augmentant le nombre de neurones cachés, au détriment de l'erreur de généralisation.

En premier lieu, nous divisons l'ensemble des observations en  $D$  sous-ensembles disjoints ; nous avons choisi  $D = 5$  qui est une valeur typique, appelée en anglais « 5-fold cross validation » [KAA 95]. Nous effectuons ensuite les étapes de calcul suivantes, pour une famille de réseaux de complexité donnée :

- 1/ Itération à réaliser  $D$  fois : construire une base d'apprentissage par l'union de  $D-1$  sous-ensembles ; effectuer plusieurs apprentissages avec des initialisations différentes des poids avec cette base d'apprentissage. Pour chacun des réseaux obtenus, calculer l'erreur quadratique moyenne sur l'ensemble de validation constituée par le sous-ensemble des observations restantes :

$$EQM = \sqrt{\frac{1}{N_v} \sum_{k=1}^{N_v} (r_k)^2} ; \text{ retenir la plus petite.}$$

- 2/ Calculer le score de validation croisée à partir des  $D$  quantités EQM :

$$SVC = \sqrt{\frac{1}{D} \sum_{i=1}^D EQM^2}$$

Ce score constitue une estimation de l'erreur de généralisation de la famille considérée.

#### IV.2.2.2. Résultats numériques :

La première partie du programme MATLAB que nous avons élaboré, nous permet de calculer les bornes inférieure et supérieure du nombre  $i$  de neurones dans la couche cachée pour chaque modèle :

*Gasoil :*

Entrée = 5 et Sortie = 1.

$$S = \sqrt{\text{entrées} * \text{sorties}} = \sqrt{5 * 1} = \sqrt{5}$$

Borne inf=  $S/2 = 1,12 \approx 1$  neurone.

Borne sup=  $S*2 = 4,47 \approx 5$  neurones.

D'où :  $i \in [1, 5]$

*Essence :*

Entrée = 4 et Sortie = 1.

$$S = \sqrt{\text{entrées} * \text{sorties}} = \sqrt{4 * 1} = 2$$

Borne inf=  $S/2 = 1$  neurone.

Borne sup=  $S*2 = 4$  neurones.

D'où :  $i \in [1, 4]$

Dans la deuxième partie, nous avons programmé la méthode de la validation croisée. Ce programme nous a permis d'entraîner 100 réseaux pour chaque famille (20 initialisations pour chacune des 5 bases d'apprentissage), avec une centaine d'itérations pour chaque initialisation, et enfin, de les tester sur différents ensembles de validation.

#### *1. Pour le gasoil :*

Pour chaque nombre de neurones cachés  $i$  ( $i \in [1, 5]$ ), le programme réalise ce qui suit :

L'ensemble des observations étant divisé en 5 sous-ensembles disjoint  $D_j$ ,  $j = 1$  à 5, il considère le sous-ensemble  $D_1$  comme ensemble de validation et construit une base d'apprentissage avec les  $D_j$  sous-ensembles restants ( $j \neq 1$ ). Avec cette base, il effectue l'apprentissage de 20 modèles avec pour chacun des initialisations différentes des poids et teste ces modèles sur le sous ensemble de validation  $D_1$ . Ceci nous donne 20 erreurs quadratiques moyennes parmi lesquelles nous choisissons la plus petite que nous avons noté EQM1.

Il applique la même procédure aux autres sous-ensembles  $D_2$ ,  $D_3$ ,  $D_4$  et  $D_5$  pour obtenir les erreurs quadratiques moyennes EQM<sub>2</sub>, EQM<sub>3</sub>, EQM<sub>4</sub> et EQM<sub>5</sub>.

Enfin, il calcule le score de validation croisée pour le nombre  $i$  de neurones cachés considéré qui constitue une estimation de l'erreur de généralisation de cette famille de réseaux (à  $i$  neurones cachés).

Voici les principaux résultats obtenus après exécution du programme :

	i = 1 neurone	i = 2 neurones	i = 3 neurones	i = 4 neurones	i = 5 neurones	Vérification i = 6
EQM1	1.0679	0.3690	0.1765	0.0208	0.0202	0.0347
EQM2	0.1010	0.0475	0.0289	0.0142	0.0094	0.0167
EQM3	0.0920	0.0717	0.0311	0.0567	0.0164	0.0226
EQM4	0.0997	0.0294	0.0543	0.0991	0.0828	0.0684
EQM5	0.1009	0.3449	0.3088	0.5770	0.1131	0.1759
SVC	0.5407	0.4153	0.3463	0.3918	<b>0.2199</b>	0.2523

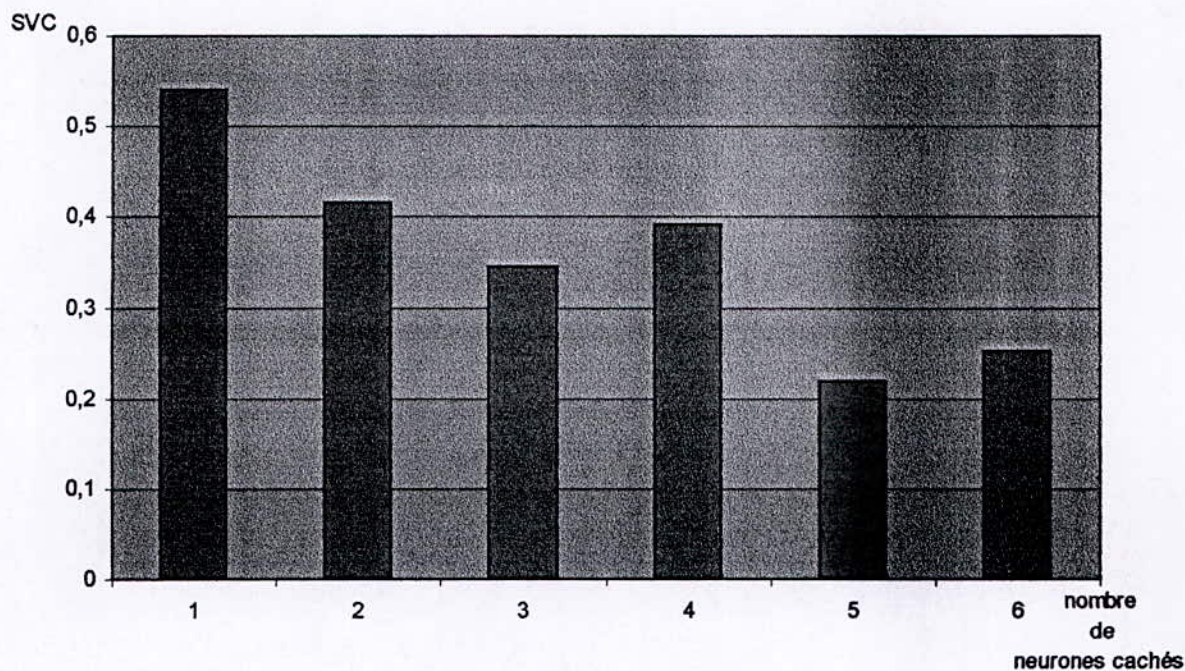
**Tableau IV.1 : Résultats de la validation croisée pour le modèle gasoil.**

Ainsi, le meilleur score de validation croisée est celui obtenu pour la famille de réseaux à 5 neurones cachés qui possède, de ce fait, la meilleure capacité de généralisation.

Bien que le nombre de neurones cachés déterminé par la méthode de la pyramide géométrique soit compris entre 1 et 5 pour le gasoil, nous avons effectué le calcul pour la famille à 6 neurones cachés pour montrer que le surajustement se traduit par une augmentation du score de validation croisée. De la sorte, la procédure peut s'arrêter lorsque cette augmentation apparaît.

Le diagramme suivant illustre l'évolution du score de validation croisée en fonction du nombre de neurones cachés :





**Figure IV.1 : Evolution du score de la validation croisée pour le modèle gasoil**

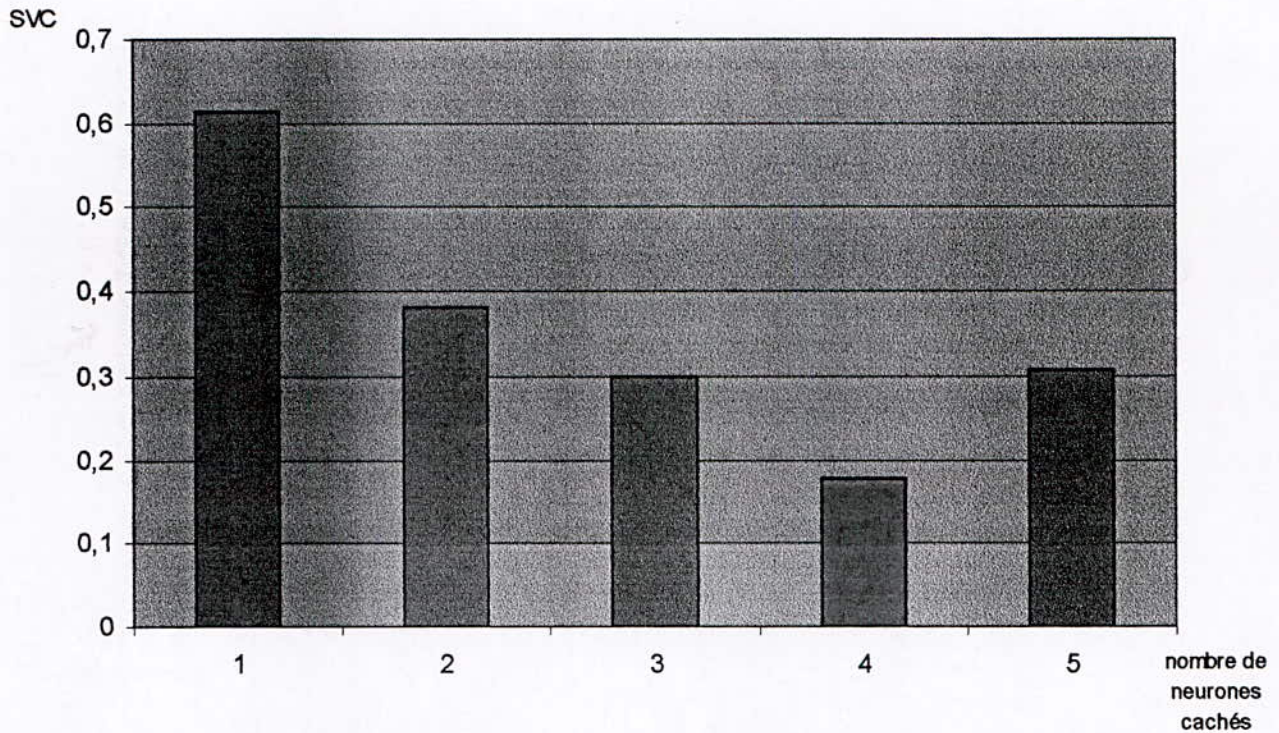
### 2. Pour l'essence :

Comme pour le modèle gasoil, nous avons exécuté le programme de la méthode de validation croisée pour le modèle essence. Les résultats obtenus sont présentés dans le tableau suivant :

	Pour i = 1	Pour i = 2	Pour i = 3	Pour i = 4	Pour i = 5
EQM1	1.1628	0.3525	0.1058	0.0069	0.0897
EQM2	0.0409	0.0225	0.0257	0.0213	0.0186
EQM3	0.1255	0.0748	0.0356	0.0496	0.0604
EQM4	0.0597	0.0238	0.0307	0.0374	0.0265
EQM5	0.4879	0.2514	0.2505	0.0404	0.2732
SVC	0.6127	0.3808	0.2995	<b>0.1764</b>	0.3061

**Tableau IV.2 : Résultats de la validation croisée pour le modèle essence.**

Ainsi, pour le modèle essence l'optimal est à 4 neurones cachés. Et toujours pour montrer le problème du surapprentissage, nous avons calculé le score de validation croisée pour la famille des réseaux à 5 neurones cachés et avons tracé l'évolution de ce score par rapport aux nombres de neurones cachés :



**Figure IV.2 : Evolution du score de la validation croisée pour le modèle essence**

#### IV.2.3. Sélection du réseau optimal :

Dans la famille des modèles performants (la famille des réseaux à 5 neurones cachés pour le modèle gasoil et 4 neurones cachés pour le modèle essence), nous allons procéder à la détermination du réseau optimal. Pour ce faire, nous effectuerons plusieurs apprentissages avec différents poids initiaux, et le critère de choix sera l'erreur de généralisation calculée sur les 3 années qui suivent la période d'apprentissage (2002-2004).

En effet, comme nous l'avons précisé à plusieurs reprises, fonder le choix du réseau sur l'erreur d'apprentissage n'est nullement significatif. Dans ce qui suit, nous allons présenter, pour chaque type de carburant, un exemple de réseau où l'erreur d'apprentissage tend vers zéro alors que les résultats de la prévision de ce même réseau sont médiocres :

- Le premier exemple est celui d'un modèle gasoil avec 5 neurones cachés. L'évolution de l'erreur d'apprentissage, mesurée par l'erreur quadratique moyenne, au cours des itérations est illustrée par le graphe suivant :

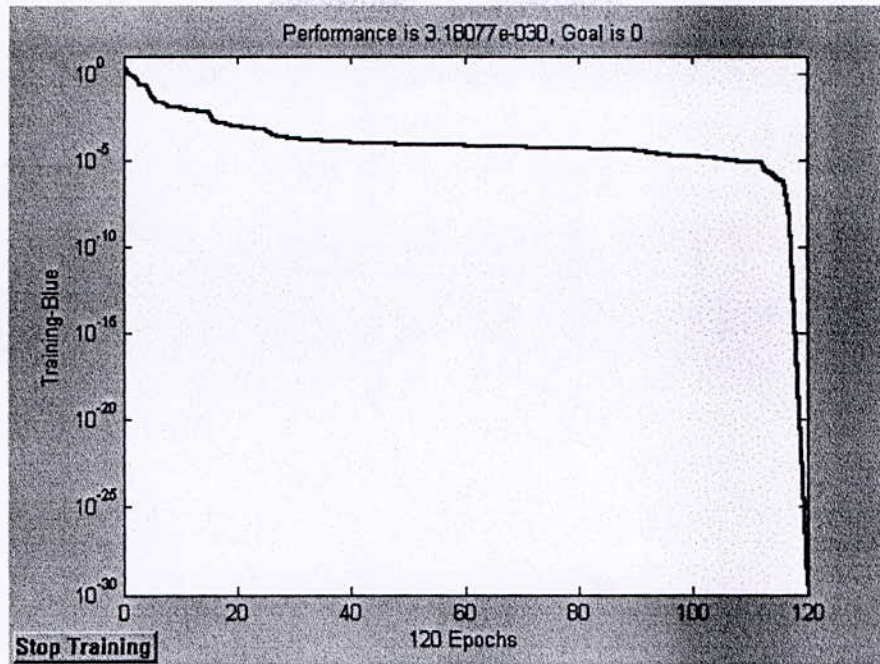


Figure IV.3 : Evolution de l'erreur d'apprentissage au cours des itérations (époques) pour le modèle gasoil.

Ainsi, avec une erreur de l'ordre de  $10^{-30}$ , les valeurs générées par le réseau sont confondues avec celles observées :

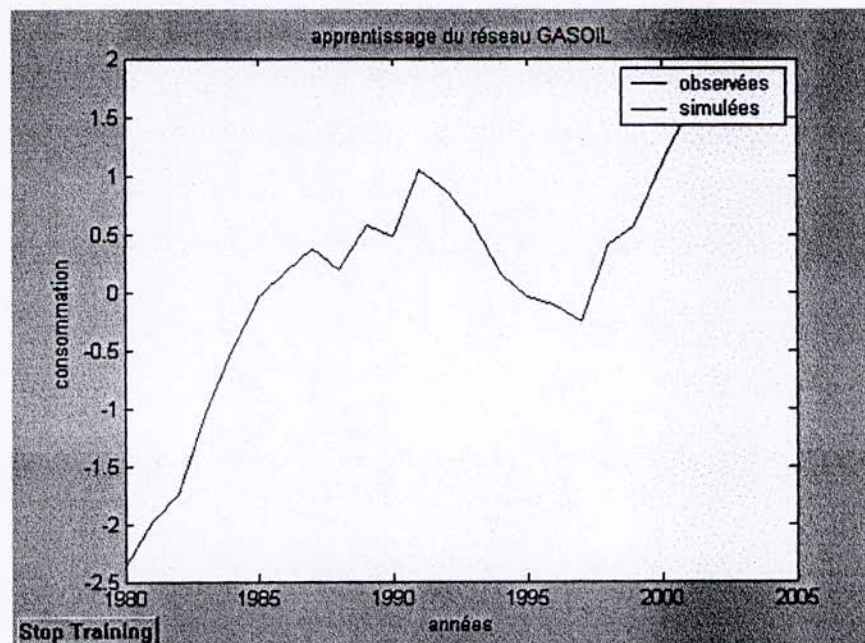
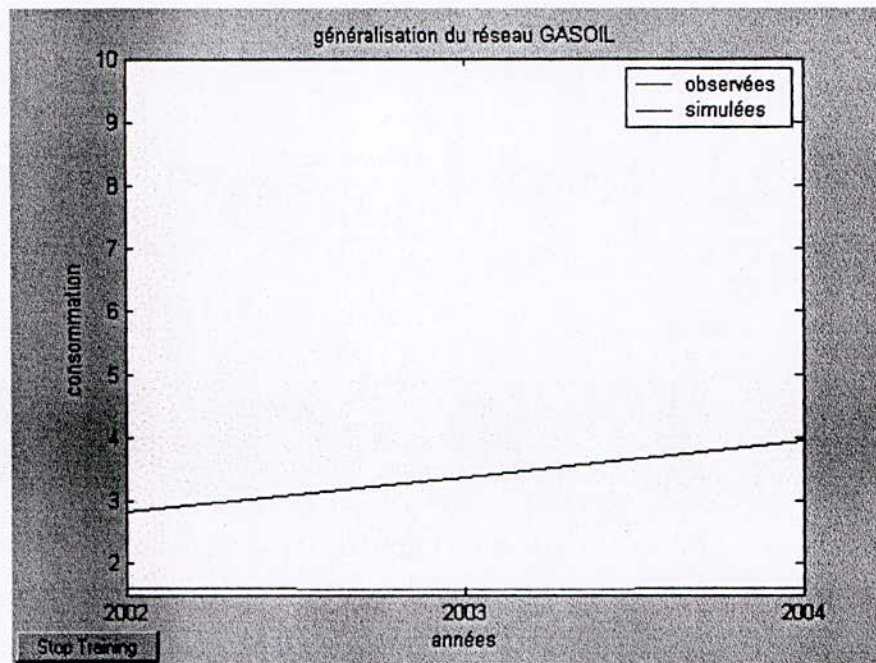


Figure IV.4 : Valeurs observées et valeurs apprises par le réseau de la consommation du gasoil de 1980 à 2001.

Mais le graphe suivant montre que lors de l'utilisation de ce même réseau pour la prévision de la consommation du gasoil pour les années 2002, 2003, 2004, les résultats ont été nettement moins bons :



**Figure IV.5 : Valeurs observées et valeurs prévues par le réseau de la consommation du gasoil de 2002 à 2004.**

- Le deuxième exemple est celui du modèle essence avec 4 neurones cachés. Ce dernier montre qu'il est tout aussi facile de concevoir un réseau dont l'erreur d'apprentissage est négligeable mais qui possède de très faibles capacités de généralisation :

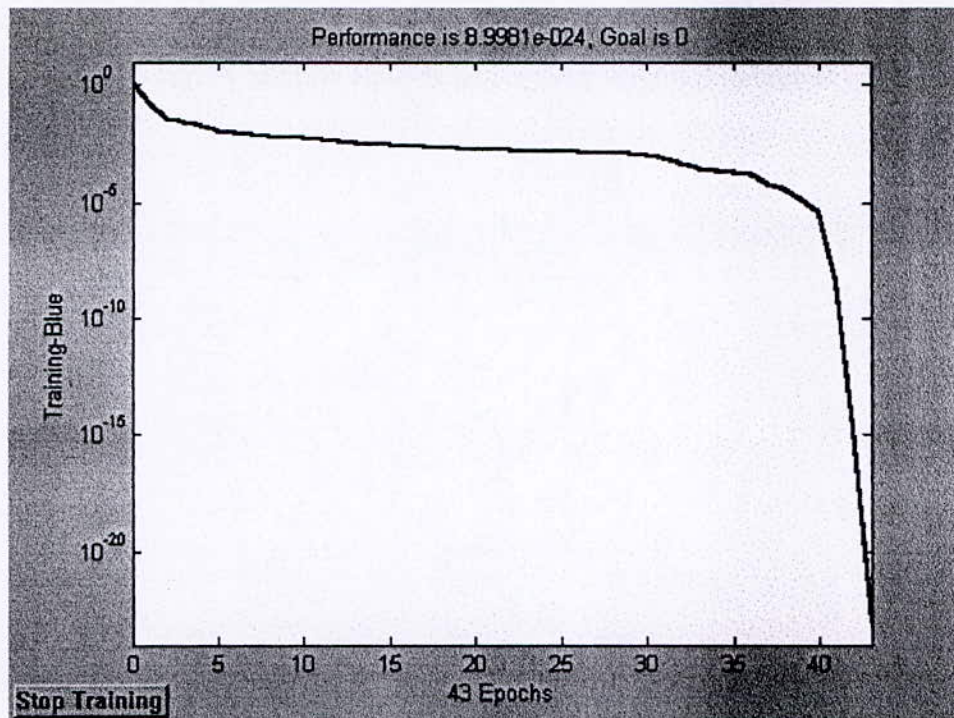


Figure IV.6 : Evolution de l'erreur d'apprentissage au cours des itérations (époches) pour le modèle essence.

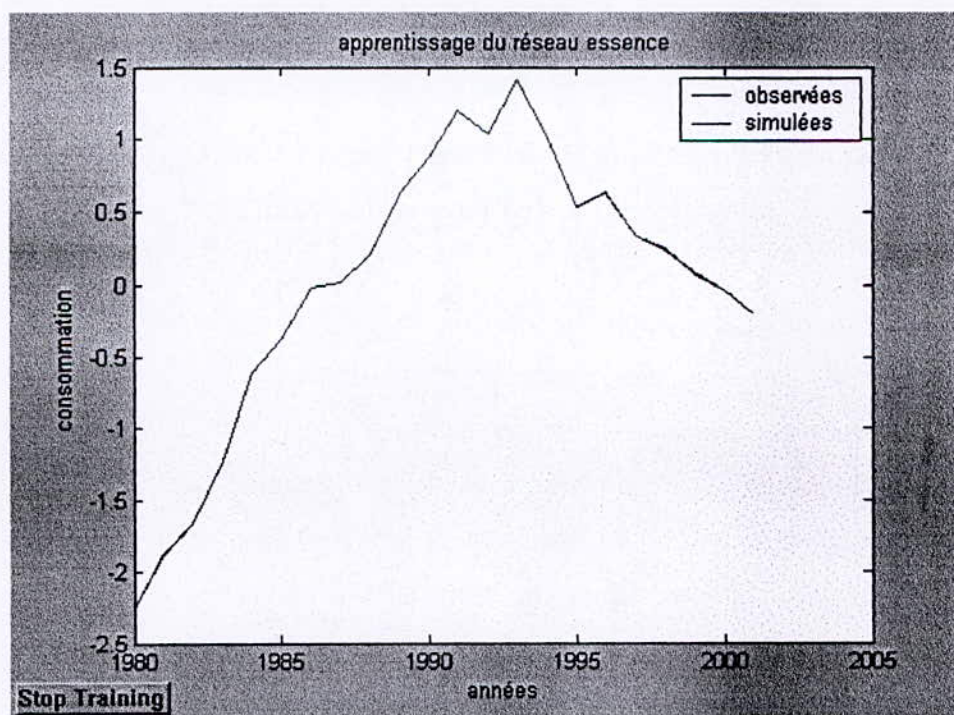
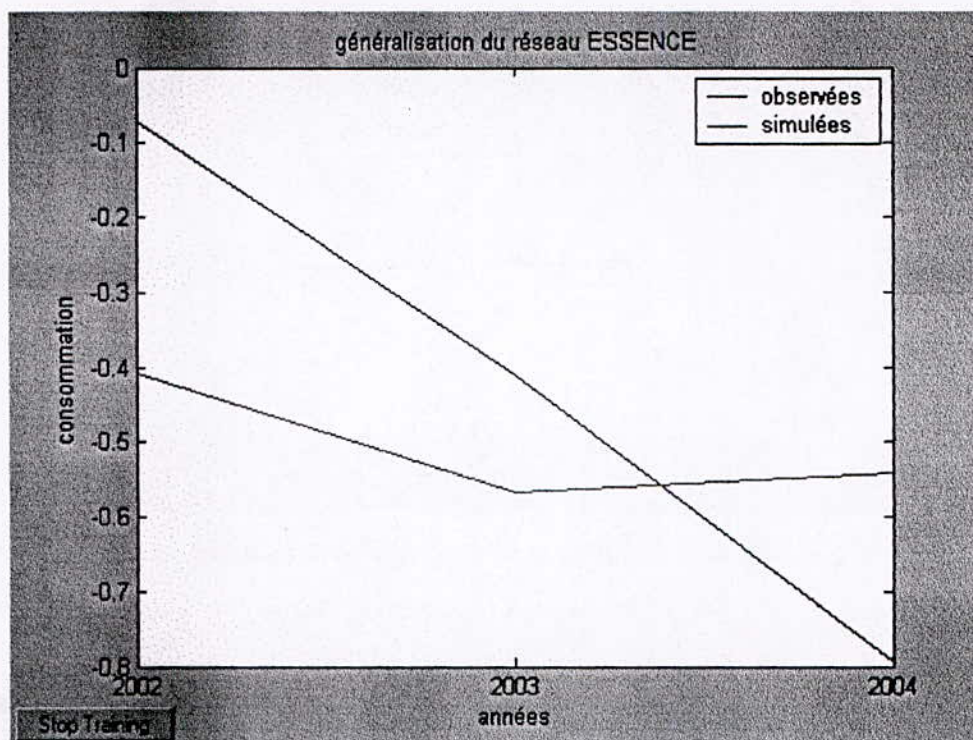


Figure IV.7 : Valeurs observées et valeurs apprises par le réseau de la consommation de l'essence de 1980 à 2001.

Nous remarquons, là aussi, que l'erreur sur les données d'apprentissage est insignifiante, ce qui prouve que le réseau a appris « par cœur ». Mais comme pour l'exemple précédent, le réseau a une très faible capacité de généralisation et donc de prévision :



**Figure IV.8 : Valeurs observées et valeurs prévues par le réseau de la consommation de l'essence de 2002 à 2004.**

Ainsi, pour le choix du réseau optimal, le programme MATLAB que nous avons élaboré dans ce but est basé sur l'erreur de généralisation. Ce programme a été exécuté plusieurs fois. Pour chaque exécution, il effectue ce qui suit :

1. L'apprentissage de 500 modèles avec des initialisations différentes des poids pour chacun d'entre eux. Ces différentes initialisations nous permettent d'obtenir plusieurs modèles avec des performances distinctes.
2. Le calcul de l'EQM de généralisation sur l'ensemble de validation pour chacun des 500 modèles et retenir la plus petite.

Ainsi, pour 5 exécutions du programme, c'est-à-dire pour 2500 réseaux différents, nous avons retenu les meilleurs EQM de chaque exécution :

Pour le gasoil :

	Exécution 1	Exécution 2	Exécution 3	Exécution 4	Exécution 5
EQM	0.1266	<b>0.0906</b>	0.1383	0.1807	0.1489

Pour l'essence :

	Exécution 1	Exécution 2	Exécution 3	Exécution 4	Exécution 5
EQM	0.0291	0.0153	0.0202	<b>0.0149</b>	0.0325

### IV.3. La prévision de la consommation énergétique :

En utilisant le modèle retenu à l'itération 2 pour le gasoil et à l'itération 4 pour l'essence, nous passons maintenant, à la phase finale qui est la prévision annuelle de la consommation des deux carburants jusqu'à l'horizon 2010 :

Pour cela, nous avons besoin des valeurs des différents facteurs pour ce même horizon. Ainsi, nous avons prévu les valeurs de ces derniers, de l'année 2005 à 2010 par les méthodes de prévision adéquates en fonction de la tendance d'évolution de chaque facteur. A cette fin, nous avons légitimement utilisé la méthode du lissage exponentiel double pour les quatre facteurs (voir les graphes ci-dessous). Dans le cas d'une évolution non linéaire d'un des facteurs, nous aurions conçu, pour la prévision des valeurs de ce dernier, un réseau de neurones dont les entrées seraient les valeurs précédentes du facteur.

Le logiciel utilisé est « STATISTICA » version 6.0 qui nous a permis, aussi, de trouver les paramètres optimaux du lissage exponentiel double (Alpha et Gamma) pour chaque facteur.

Dans ce qui suit, nous présentons les résultats du lissage des différents facteurs comparés à leurs valeurs observées respectives en y associant l'erreur quadratique moyenne (cf. Annexe 6) :

La population urbaine :

Alpha	Gamma	EQM
0,93	0,497	0,00031601

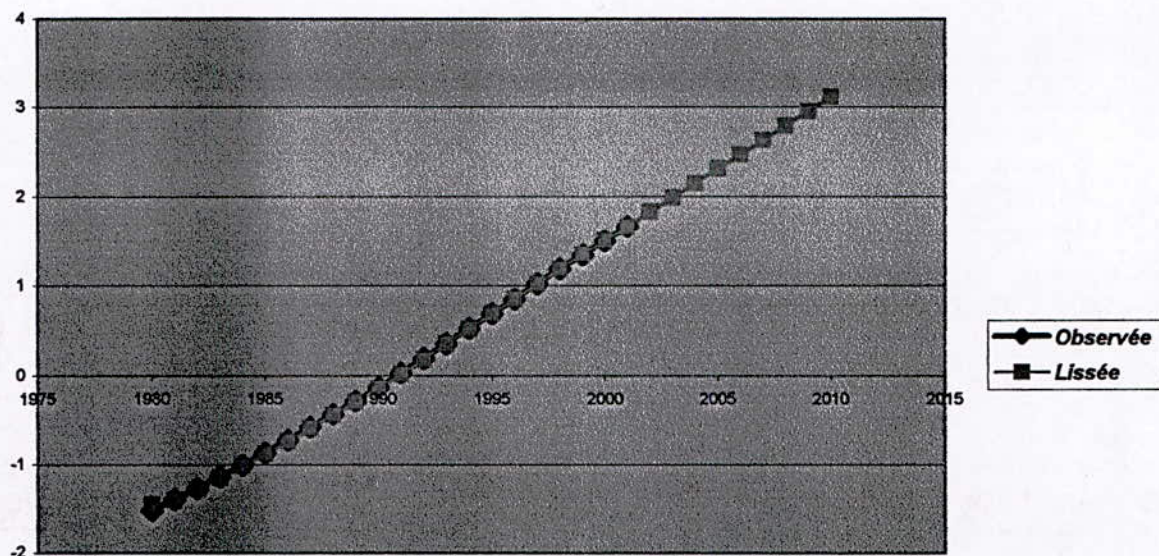


Figure IV.9 : Valeurs observées et lissées de l'évolution de la population urbaine

La population :

Alpha	Gamma	EQM
1	0,478	0,00046041

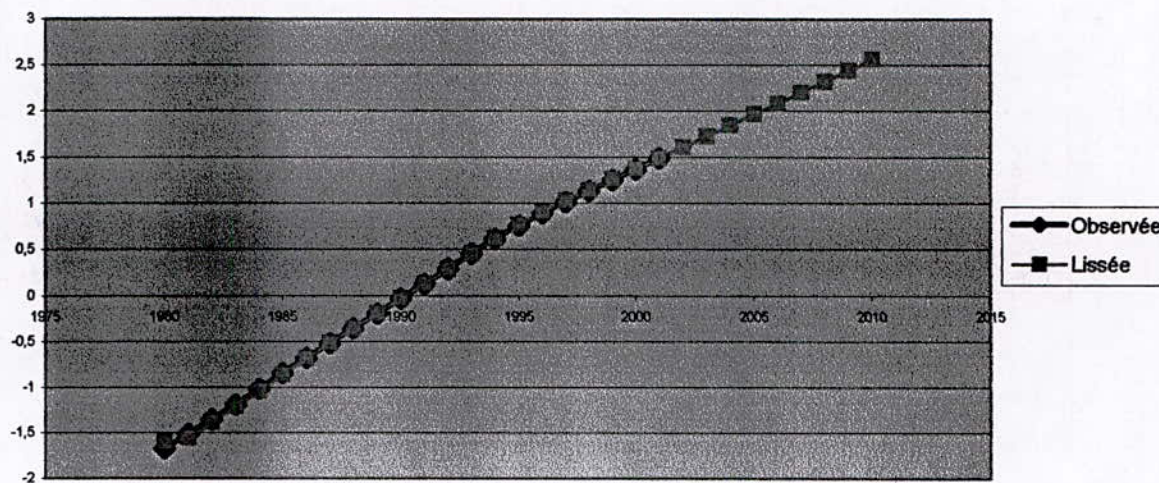


Figure IV.10 : Valeurs observées et lissées de l'évolution de la population

Le parc automobile :

Alpha	Gamma	EQM
0,581	1	0,00486243



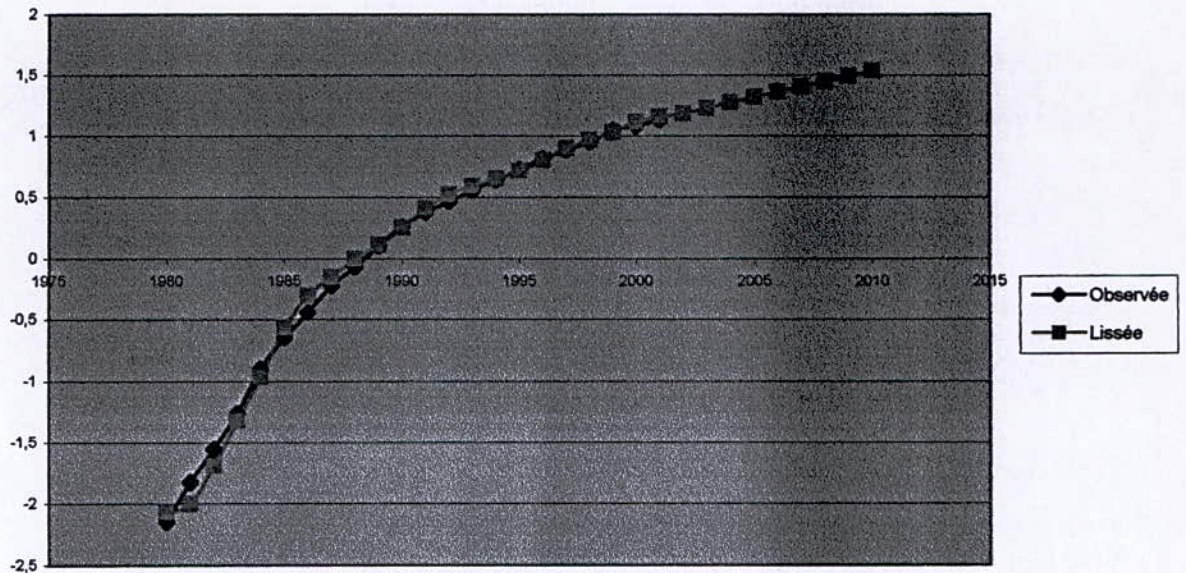


Figure IV.11 : Valeurs observées et lissées de l'évolution du parc auto

Les dépenses du gouvernement pour la consommation finale :

Alpha	Gamma	EQM
1	0	0,03546387

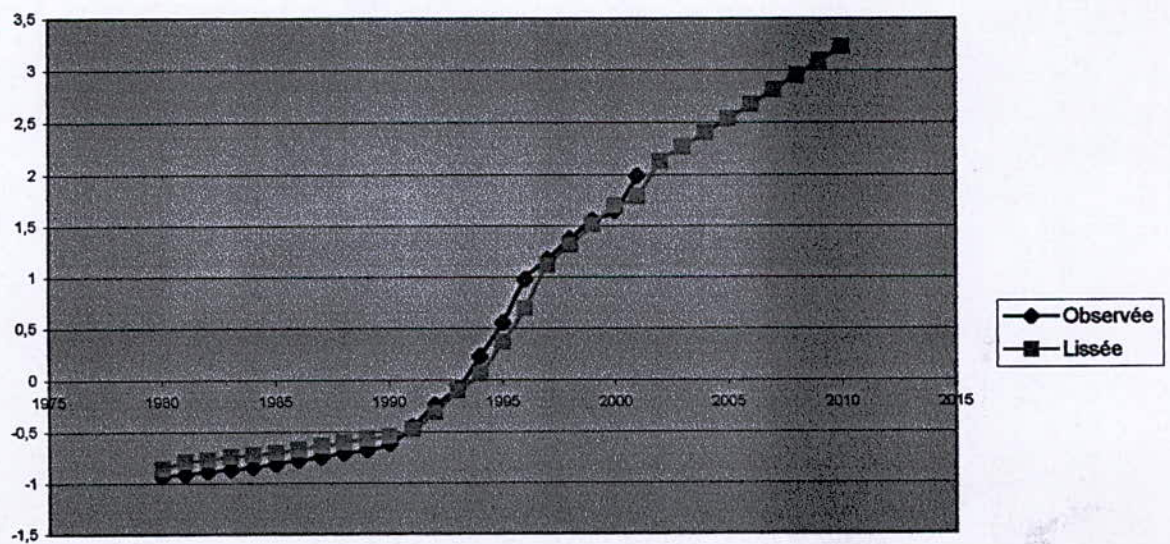


Figure IV.12 : Valeurs observées et lissées des dépenses du gouvernement

Enfin, nous introduirons à chaque itération, en plus de ces facteurs, la valeur de la consommation prévue à l'itération précédente.

Toutes les données relatives aux variables d'entrée étant disponibles et le réseau optimal sélectionné, nous pouvons, à présent, exécuter le programme MATLAB élaboré pour la prévision. Les résultats obtenus (toujours centrés réduits) sont les suivants :

Année	Prévision gasoil	Prévision essence
2002	2,3184	-0,2501
2003	3,152	-0,3029
2004	3,7708	-0,6205
2005	4,0976	-0,7368
2006	4,2554	-0,685
2007	4,344	-0,6099
2008	4,4023	-0,5178
2009	4,4436	-0,4372
2010	4,4737	-0,3743

**Tableau IV.3 : Résultats de la prévision (centrés réduits).**

A partir de ces résultats, nous allons déduire les valeurs réelles de la consommation en multipliant les données centrées réduites, ci-dessus, par l'écart type et en ajoutant la moyenne à ce produit :

Année	Prévision gasoil (tonne)	Prévision essence (tonne)
2002	4307267,862	1824011,912
2003	4722965,439	1807874,278
2004	5031547,086	1710803,965
2005	5194514,898	1675258,381
2006	5273206,210	1691090,378
2007	5317389,038	1714043,717
2008	5346461,937	1742192,886
2009	5367057,319	1766827,228
2010	5382067,512	1786051,796

**Tableau IV.4 : Résultats de la prévision.**

Les graphes associés à la prévision de l'évolution de chaque carburant sont présentés ci-dessous :

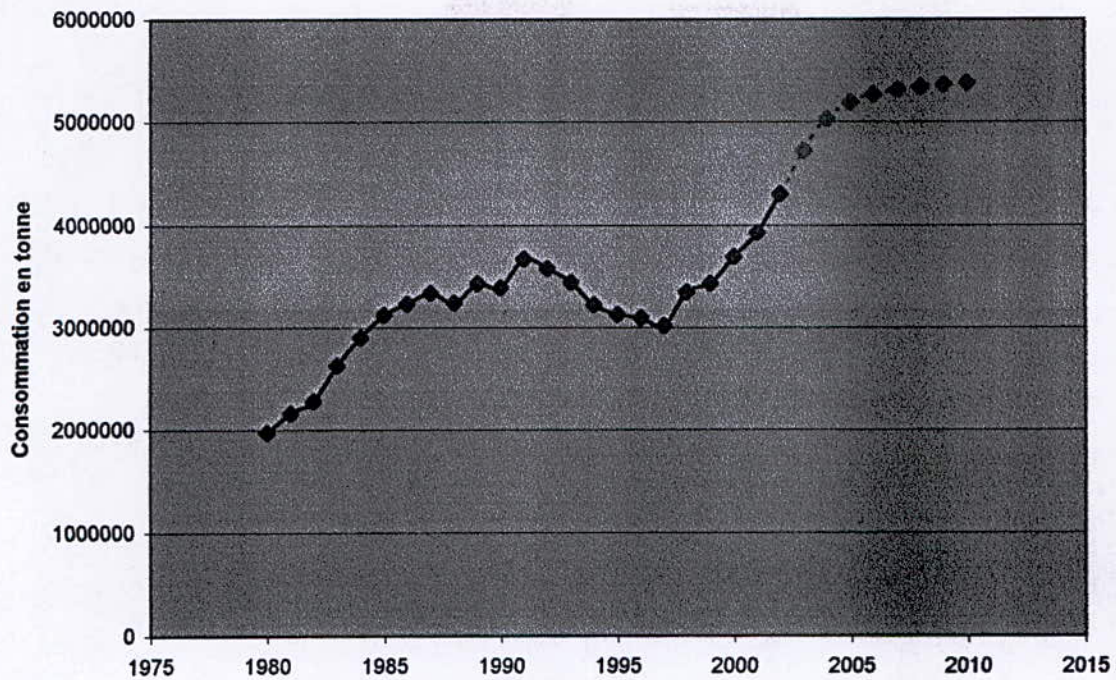


Figure IV.13 : Prévision de la consommation du gasoil

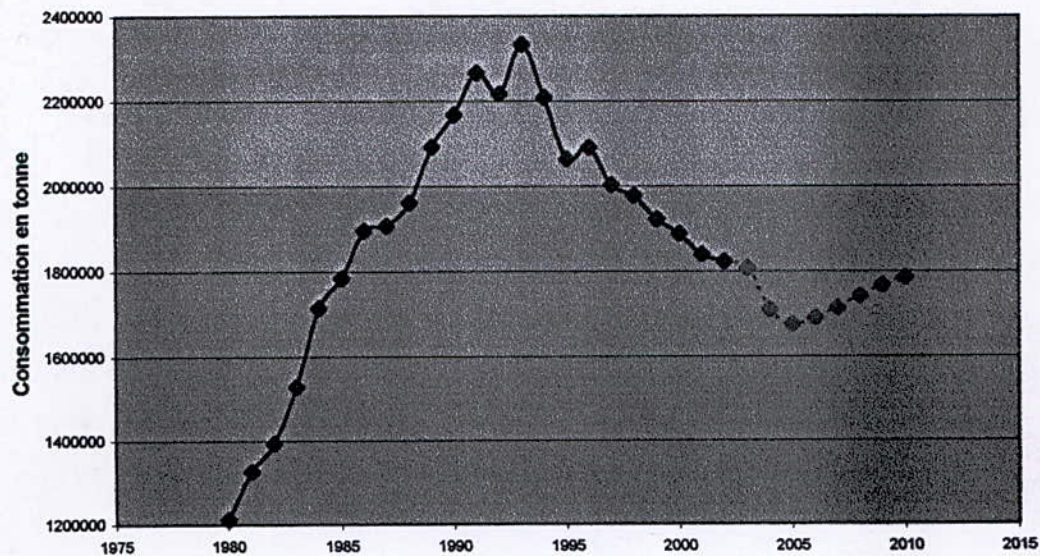


Figure IV.14 : Prévision de la consommation de l'essence

#### IV.4. Calcul d'erreurs :

Le calcul d'erreurs est un critère de sélection et de comparaison entre les modèles d'un côté et une façon de fournir une information plus précise et plus utile d'un autre côté.

Ainsi, Pour les résultats ci-dessus, nous nous sommes basés sur les deux statistiques qui sont les plus utilisées, à savoir :

- MSE (EQM) : Erreur Quadratique Moyenne.
- MAPE (EAMP) : Erreur Absolue Moyenne en Pourcentage.

La première statistique, utilisée par la validation croisée, représente un bon critère de comparaison sur lequel nous avons fondé notre choix du réseau optimal pour chaque carburant et dont les valeurs réelles (non centrées réduites) sont :

$$\left\{ \begin{array}{l} \text{MSE}_{\text{Gasoil}} = \frac{1}{3} \sum_{i=1}^3 e_i^2 = 0.0906 * \text{Ecart type}_{\text{Gasoil}} = 45180.2 \text{ Tonnes.} \\ \text{MSE}_{\text{Essence}} = \frac{1}{3} \sum_{i=1}^3 \frac{|e_i|}{d_i} = 0.0149 * \text{Ecart type}_{\text{Essence}} = 4554 \text{ Tonnes.} \end{array} \right.$$

Avec :

$e$  = erreur de généralisation.

Il est clair qu'une deuxième statistique, qui permet d'avoir une idée relative et donc plus représentative de l'erreur commise, est nécessaire. Il s'agit, dans notre étude, de l'Erreur Absolue Moyenne en Pourcentage :

$$\left\{ \begin{array}{l} \text{MAPE}_{\text{Gasoil}} = 3.22 \% \\ \text{MAPE}_{\text{Essence}} = 2.66 \% \end{array} \right.$$

Nous remarquons que les Erreurs Absolues Moyennes en Pourcentage pour les deux modèles ne dépassent pas les 10 %, ce qui nous conduit à dire que les résultats de prévision sont satisfaisants. [MEL, 90]

# *Conclusion générale*

## *Conclusion générale :*

L'introduction des réseaux de neurones artificiels (RNA) dans le domaine de la prévision et les progrès observés ces dernières années ne constituent probablement que les premiers balbutiements d'une nouvelle manière de conceptualiser les phénomènes économiques. Ces outils puissants de traitement non linéaire des données n'ont pas encore atteint leur plein développement pour des raisons liées à leur relative nouveauté dans le monde industriel.

Ces réseaux, et spécialement les Perceptron Multi Couches que nous avons utilisés, ont bénéficié d'une attention particulière grâce à leurs nombreux avantages par rapport aux méthodes de modélisation conventionnelles dont la pertinence est de plus en plus contestée, essentiellement à cause de leurs hypothèses simplificatrices.

En effet, les RNA contribuent largement au traitement adéquat des décisions dites « non structurées ». Cette contribution se manifeste par la diminution du degré d'incertitude ou encore par un traitement judicieux de la complexité des phénomènes observés.

Ces outils intelligents ont suscité en nous une curiosité et un intérêt qui se sont traduits par l'élaboration du présent travail. Ils nous ont permis de concevoir un modèle pour la prévision de la consommation des carburants (gasoil et essence) à l'horizon 2010, qui se prête tout à fait à ce type d'applications vu son comportement non linéaire.

Pour cela nous avons adopté la démarche suivante :

- Nous familiariser avec ce domaine, à la fois complexe et passionnant, en nous intéressant d'abord aux propriétés fondamentales qui justifient l'intérêt croissant qui lui est accordé et qui ne cesse de croître devant les non linéarités qui abondent dans le monde économique.

- Présenter une méthodologie robuste pour le prétraitement des données fondée sur l'algorithme de Gram-Schmidt et le test de Fisher pour le classement et la sélection des descripteurs. Cette méthodologie peut être utilisée par n'importe quelle approche de prévision.

- Présenter une procédure qui permet la conception et le choix du réseau neuronal optimal qui est la validation croisée.

- Concevoir le modèle en utilisant le logiciel MATLAB, qui a l'avantage d'offrir une boîte à outils permettant l'apprentissage et la conception des RNA. De même que nous avons été amené à élaborer les programmes pour la mise en œuvre des éléments de méthodologie

qui ne sont pas spécifiquement neuronaux tels que la méthode de la validation croisée, de Gram Schmidt, ...

Ce modèle a finalement été appliqué pour la prévision de la consommation des carburants qui n'était jusqu'alors abordé que par des méthodes traditionnelles, et les résultats obtenus sont encourageants dès lors que l'erreur commise ne dépasse pas les 10%. Il constitue un outil d'aide à la décision intéressant et pourra être mis à la disposition des managers en vue de l'élaboration de prévision plus fiables.

Néanmoins, les RNA possèdent quelques limites à savoir :

- Les RNA restent encore, en partie, une boîte noire car ils ne permettent pas d'expliquer les relations pertinentes entre les variables. De ce point de vue, le système n'a donc qu'un pouvoir explicatif médiocre et n'est pas capable de retracer le cheminement suivi pour atteindre le résultat.
- Les modèles les plus simples sont souvent les meilleurs et les moins coûteux en temps de calcul, donc les RNA pourraient être utiles lorsque les modèles simples produisent des résultats insatisfaisants ce qui est généralement le cas des modèles économiques.
- Les RNA sont utilisés comme des outils de modélisation par apprentissage, qui permettent d'ajuster des fonctions non linéaires très générales à des ensembles de points ; comme toute méthode qui s'appuie sur des techniques statistiques, l'utilisation des RNA nécessite que l'on dispose de données suffisamment représentatives.

Ces limites ne constituent pas pour autant un grand handicap pour les RNA dont les applications en finance et en économie continuent de faire surface. D'autant plus que, d'une part, tous ces problèmes font aujourd'hui l'objet de recherches dont nous pouvons espérer qu'elles apportent des réponses satisfaisantes dans les années qui viennent, et d'autre part, différentes recherches s'intéressent à des techniques hybrides évolutives qui combinent les RNA avec d'autres approches telles que les algorithmes génétiques (AG) connus pour leur puissance en matière d'optimisation ce qui permet d'obtenir une meilleure performance.

Enfin, il convient de dire que même si l'outil est intéressant, il ne pourra jamais, comme au demeurant toutes les méthodes de prévision, prévoir l'imprévisible.

*Annexes*



# ANNEXE 1 :

## Généralités sur les RNA

### 1.1. Historique : [PAR 04], [TOU 92]

Les premières recherches remontent à la fin du 19<sup>ème</sup> et au début du 20<sup>ème</sup> siècle. Elles consistent en de travaux multidisciplinaires en physique, en psychologie et en neurophysiologie. A cette époque, il s'agissait de théories plutôt générales sans modèle mathématique précis d'un neurone. La naissance du domaine des réseaux de neurones artificiels remonte aux années 1940 avec les travaux de *Warren McCulloch* et *Walter Pitts* qui ont montré qu'avec de tels réseaux, on pouvait, en principe, calculer n'importe quelle fonction arithmétique ou logique. Vers la fin des années 1940, *Donald Hebb*, physiologiste américain, a ensuite proposé une théorie fondamentale pour l'apprentissage.

La première application concrète des réseaux de neurones artificiels est survenue vers la fin des années 1950 avec l'invention du réseau dit «perceptron» par *Frank Rosenblatt* et ses collaborateurs qui ont construit un réseau et démontré ses habilités à reconnaître des formes. Malheureusement, il a été démontré par la suite que ce perceptron simple ne pouvait résoudre qu'une classe limitée de problème. Environ au même moment, *Bernard Widrow* et *Ted Hoff* ont proposé un nouvel algorithme d'apprentissage pour entraîner un réseau adaptatif de neurones linéaires (Adaline), dont la structure et les capacités sont similaires au perceptron.

Vers la fin des années 1960, un livre publié par *Marvin Minsky* et *Seymour Papert* est venu jeter beaucoup d'ombre sur le domaine des réseaux de neurones. Ces deux auteurs ont démontré les limitations des réseaux développés par Rosenblatt, Widrow et Hoff. Ils ont conclu à tort que le domaine des réseaux de neurones était sans issue et qu'il fallait cesser de s'y intéresser et de financer la recherche dans ce domaine.

En 1972, *Teuvo Kohonen* et *James Anderson* ont développé indépendamment et simultanément d'autres types de réseaux avec de nouvelles architectures et de nouveaux algorithmes plus puissants.

En 1982, *J.J.Hopfield* présente une théorie du fonctionnement et des possibilités des réseaux de neurones. C'est à lui qu'on doit le regain d'intérêt pour les réseaux de neurones artificiels

Vers la fin de l'année 1985, une pierre d'achoppement a été levée par l'algorithme de « rétropropagation des erreurs ». Cet algorithme est la réponse aux critiques de Minsky et

Papert formulées à la fin des années 1960. C'est ce nouveau développement, attribué à *David Rumelhart* et *James McClelland*, qui a littéralement ressuscité le domaine des réseaux de neurones. Depuis ce temps, c'est un domaine où bouillonnent constamment de nouvelles théories, de nouvelles structures et de nouveaux algorithmes.

## 1.2. Fondement biologique :

### 1.2.1. Morphologie sommaire du neurone :

Le neurone est une cellule très spécialisée, responsable du traitement et de la propagation des messages nerveux. En effet, en fonction des informations qu'elle reçoit, elle est capable d'émettre des signaux électriques, de les propager le long de ses prolongements et de les transmettre par l'intermédiaire de connexions, ou synapses. Il est observé qu'une cellule est connectée, en moyenne, à 1000 voire 10 000 autres cellules. On estime leur nombre moyen, chez un être humain, à environ  $10^{11}$ . Ce nombre est comparable à celui avancé pour la totalité des étoiles de notre galaxie. On en déduit que le nombre total des synapses est de l'ordre de  $10^{14}$  à  $10^{15}$ . [FAU 98]

### 1.2.2. Structures des neurones :

Bien qu'un grand nombre de neurones existe, il est commode de retenir que le neurone « modèle » est composé de trois parties :

- le corps cellulaire (soma).
- Les dendrites.
- L'axone.

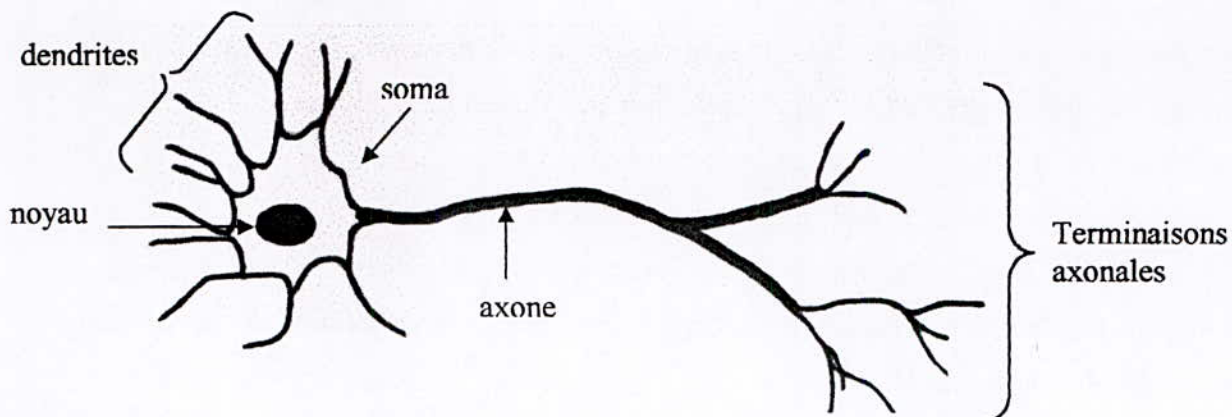


Figure 1.1 : Schématisation d'un neurone. [ARB 03]

#### 1. Le corps cellulaire (soma) :

Le corps cellulaire ou soma est la partie du neurone qui contient le noyau cellulaire et le cytoplasme qui l'entoure. Il traite les signaux électriques qui lui proviennent de ses dendrites.

Au cas où ces signaux dépassent un certain seuil, le soma transmet un courant électrique résultant du traitement aux neurones auxquels il est connecté par l'intermédiaire de son axone.

[RIC 01]

### 2. Les dendrites :

Les dendrites et le soma reçoivent de très nombreux contacts synaptiques d'autres neurones et constituent de ce fait la principale surface de réception du neurone. Elles génèrent, en réponse à ces messages, des signaux électriques et intègrent ces messages afférents. [HAM 90]

### 3. L'axone :

L'axone n'est pas un prolongement unique, il se divise en une ou plusieurs collatérales d'axone qui forment les terminaisons axonales. Il est entouré d'une gaine de myéline, ayant la propriété d'isoler électriquement l'axone

Les potentiels d'action se propagent le long de l'axone jusqu'aux terminaisons axonales. En réponse à l'arrivée des potentiels d'action, les terminaisons axonales libèrent ou non les neurotransmetteurs qu'elles renferment. Cette libération est localisée précisément au niveau des contacts synaptiques. Pour toutes ces raisons, l'axone est considéré comme le pôle émetteur du neurone. [HAM 90]

#### 1.2.3. L'influx nerveux :

L'information nerveuse est véhiculée sous forme d'influx, à l'issue de l'excitation des cellules nerveuses. Cette excitation peut être d'une nature quelconque : mécanique, électrique, chimique, etc. Cet influx est de nature électro-ionique, il est constitué d'impulsions. Les influx afférents, transmis par les synapses réceptrices réparties sur les dendrites et le corps cellulaire, développent un potentiel électrique traité dans la cellule. Quand le potentiel atteint un seuil, le neurone émet un signal appelé potentiel d'action, qui est transmis, sans atténuation par l'axone et ses terminaisons sur les autres neurones. [ARB 03]

#### 1.2.4. Les synapses :

Dès 1888, on établit que les contacts entre les terminaisons axonales d'un neurone et les dendrites ou le soma d'un autre neurone, sont les lieux de passage de l'influx nerveux. Le terme « synapse » a été proposé par la suite par Sherrington en 1897, qui veut dire « j'attache » en grec, pour désigner ces zones de contacts entre neurones, spécialisées dans la transmission d'informations. [FAU 98]

On distingue, selon des critères morphologiques et fonctionnels, plusieurs types de synapses parmi lesquelles : Les synapses chimiques et Les synapses électriques. [HAM 90]

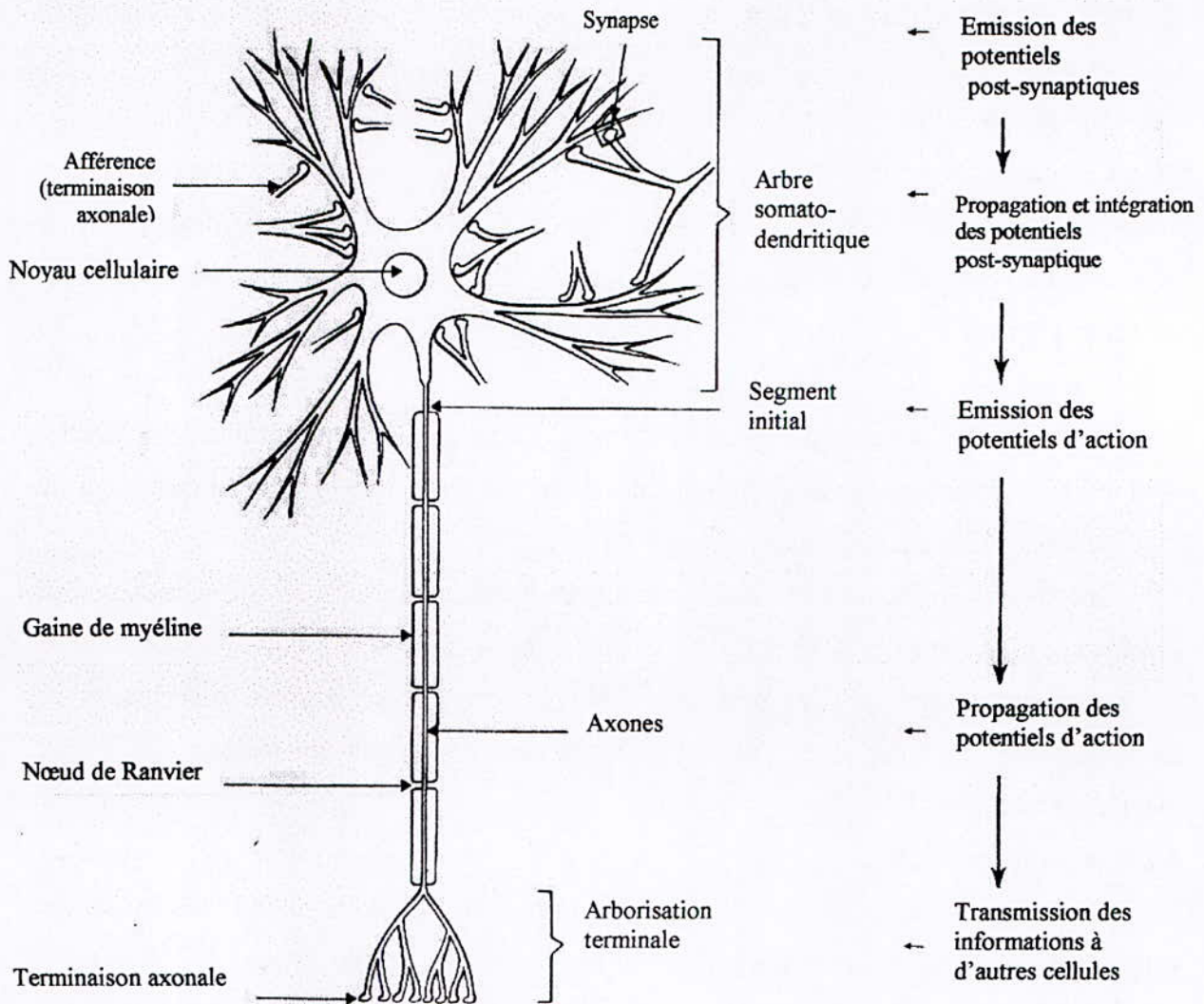


Figure 1.2 : Schéma récapitulatif du fonctionnement d'un neurone. [RIC 01]

### 1.2.5. L'organisation des neurones en réseaux :

Un neurone pris isolément, malgré la complexité du traitement qu'il effectue, ne peut prendre en charge la totalité du contrôle d'un être vivant évolué. Sa structure lui permet d'entrer en relation avec des milliers d'autres neurones où l'axone se ramifie pour diffuser le résultat de la décision à d'autres neurones. Par des contacts multiples, c'est un réseau de connexions très dense qui se construit, et qui permet un échange riche entre des centaines de neurones différents. Il est ainsi possible de mettre en évidence, dans le système nerveux, des structures organisées, qui diffèrent selon la nature de l'information recueillie et selon le traitement effectué. [DAV 92]

## ANNEXE 2 :

### *Les règles d'apprentissage*

#### 2.1. La règle de Hebb :

*«When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic changes take place in one or both cells such that A's efficiency as one of the cells firing B, is increased.»*

Donald HEBB. [TOU 92]

Dans un contexte neurobiologique, Hebb cherchait à établir une forme d'apprentissage associatif au niveau cellulaire. Dans le contexte des réseaux artificiels, on peut reformuler l'énoncé de Hebb sous la forme d'une règle d'apprentissage en deux parties :

1. Si deux neurones de part et d'autre d'une synapse (connexion) sont activés simultanément (d'une manière synchrone), alors la force de cette synapse doit être augmentée.
2. Si les deux neurones sont activés d'une manière asynchrone, alors la synapse correspondante doit être affaiblie ou carrément éliminée. [TOU 92]

Mathématiquement, on peut exprimer la règle de Hebb sous sa forme la plus simple par la formule suivante :

$$w_j(t) = w_j(t-1) + \Delta w_j(t-1) \quad (2.1)$$

Avec : 
$$\Delta w_j(t-1) = \eta p_j(t)a(t) \quad (2.2)$$

Où  $\eta$  est une constante positive qui détermine la vitesse de l'apprentissage,  $p_j(t)$  correspond à l'activité pré-synaptique (l'entrée  $j$  du neurone) au temps  $t$ , et  $a(t)$  à l'activité post-synaptique (sortie du neurone) à ce même temps  $t$ . Cette formule fait ressortir explicitement la corrélation entre le signal qui entre et celui qui sort. Sous une forme vectorielle, on écrit :

$$\Delta \mathbf{w}(t-1) = \eta \mathbf{p}(t)\mathbf{a}(t) \quad (\text{II.7})(2.3)$$

Un problème immédiat avec la règle de l'équation 2.3 est que les changements de poids  $\Delta w_j(t)$  peuvent croître de façon exponentielle si, par exemple, l'entrée et la sortie demeurent constantes dans le temps. Pour pallier à cette croissance exponentielle qui provoquerait

invariablement une saturation du poids, on ajoute parfois un facteur d'oubli qui retranche de la variation de poids, une fraction  $\alpha$  du poids actuel. On obtient ainsi :

$$\Delta w_j(t-1) = \eta p_j(t)a(t) - \alpha w_j(t-1) \quad (2.4)$$

Où  $0 \leq \alpha \leq 1$  est une nouvelle constante. Sous forme vectorielle, on écrit :

$$\Delta \mathbf{w}(t-1) = \eta \mathbf{p}(t)\mathbf{a}(t) - \alpha \mathbf{w}(t-1) \quad (2.5)$$

Mais cette règle ne résout pas tous les problèmes. A cause du terme d'oubli, il est primordial que les stimuli soient répétés régulièrement, sinon les associations apprises grâce à la règle de l'équation 2.5 seront éventuellement perdues car complètement oubliées. Une autre variante de la règle de Hebb s'exprime donc de la manière suivante :

$$\Delta w_j(t-1) = \eta p_j(t)a(t) - \alpha a(t)w_j(t-1) \quad (2.6)$$

Et si l'on fixe  $\alpha = \eta$  pour simplifier (on impose un rythme d'apprentissage égal à celui de l'oubli), on obtient la règle dite «instar» :

$$\Delta w_j(t-1) = \eta a(t)[p_j(t) - w_j(t-1)] \quad (2.7)$$

Que l'on peut réécrire sous forme vectorielle de la façon suivante :

$$\Delta \mathbf{w}(t-1) = \eta \mathbf{a}(t) [\mathbf{p}(t) - \mathbf{w}(t-1)] \quad (2.8)$$

## 2.2. Règle de correction des erreurs :

Cette règle est fondée sur la correction de l'erreur observée en sortie. Soit  $a_i(t)$  la sortie que l'on obtient pour le neurone  $i$  au temps  $t$ . Cette sortie résulte d'un stimulus  $\mathbf{p}(t)$  que l'on applique aux entrées du réseau dont un des neurones correspond au neurone  $i$ . Soit  $d_i(t)$  la sortie que l'on désire obtenir pour ce même neurone  $i$  au temps  $t$ . Alors,  $a_i(t)$  et  $d_i(t)$  seront généralement différents et il est naturel de calculer l'erreur  $e_i(t)$  entre ce qu'on obtient et ce que l'on voudrait obtenir et de chercher un moyen de réduire autant que possible cette erreur :

$$e_i(t) = d_i(t) - a_i(t) \quad (2.9)$$

Sous forme vectorielle, on obtient :

$$\mathbf{e}(t) = \mathbf{d}(t) - \mathbf{a}(t) \quad (2.10)$$

Avec  $\mathbf{e}(t) = [e_1(t), e_2(t) \cdots e_i(t) \cdots e_S(t)]$  qui désigne le vecteur des erreurs observées sur les  $S$  neurones de sortie du réseau. L'apprentissage par correction des erreurs consiste à minimiser un indice de performance  $F$  basé sur les signaux d'erreur  $e_i(t)$ , dans le but de faire converger les sorties du réseau avec ce qu'on voudrait qu'elles soient. Un critère très utilisé est celui de la somme des erreurs quadratiques :

$$F(e(t)) = \sum_{i=1}^S e_i^2(t) = e(t)^T e(t) \quad (2.11)$$

Il importe de remarquer que les paramètres libres d'un réseau sont ses poids. Prenons l'ensemble de ces poids et assemblons les sous la forme d'un vecteur  $w(t)$  au temps  $t$ . Pour minimiser  $F(e(t)) = F(w(t)) = F(t)$ , nous allons commencer par choisir des poids initiaux ( $t = 0$ ) au hasard, puis nous allons modifier ces poids de la manière suivante :

$$w(t + 1) = w(t) + \eta x(t) \quad (2.12)$$

Où le vecteur  $x(t)$  désigne la direction dans laquelle nous allons chercher le minimum et  $\eta$  est une constante positive déterminant l'amplitude du pas dans cette direction (la vitesse d'apprentissage). L'objectif est de faire en sorte que  $F(t + 1) < F(t)$ . Pour choisir la direction  $x$  tout en respectant la condition précédente, nous allons considérer la série de Taylor de 1<sup>er</sup> ordre autour de  $w(t)$  :

$$F(t + 1) = F(t) + \nabla F(t)^T \Delta w(t) \quad (2.13)$$

Où  $\nabla F(t)$  désigne le gradient de  $F$  par rapport à ses paramètres libres (les poids  $w$ ) au temps  $t$ , et  $\Delta w(t) = w(t + 1) - w(t)$ . Pour que  $F(t + 1) < F(t)$ , il faut que la condition suivante soit respectée :

$$\nabla F(t)^T \Delta w(t) = \eta \nabla F(t)^T x(t) < 0 \quad (2.14)$$

N'importe quel vecteur  $x(t)$  qui respecte l'inégalité de l'équation II.18 s'oriente donc dans une direction qui diminue  $F$ . On parle alors d'une direction de «descente». Pour obtenir une descente maximum, étant donnée  $\eta > 0$ , il faut que le vecteur  $x(t)$  aille dans le sens opposé au gradient car c'est dans ce cas que le produit scalaire sera minimum :

$$x(t) = - \nabla F(t) \quad (2.15)$$

Ce qui engendre la règle dite de «descente du gradient» :

$$\Delta w(t) = - \eta \nabla F(t) \quad (2.16)$$

L'expression exacte du gradient dépend de la fonction d'activation utilisée pour les neurones. La règle de la correction des erreurs est utilisée pour beaucoup de réseaux de neurones artificiels, bien qu'elle ne soit pas plausible biologiquement. En effet, comment le cerveau pourrait-il connaître a priori les sorties qu'il doit produire ? Cette règle ne peut être utilisée que dans un contexte d'apprentissage supervisé. [PAR 04]

### 2.3. Règle d'apprentissage compétitif :

L'apprentissage compétitif consiste à mettre en compétition les neurones d'un réseau pour déterminer lequel sera actif à un instant donné. Contrairement aux autres types d'apprentissage où, généralement, tous les neurones peuvent apprendre simultanément et de la même manière, l'apprentissage compétitif produit un «vainqueur» ainsi que, parfois, un ensemble de neurones «voisins» du vainqueur, et seuls ce vainqueur et, potentiellement, son voisinage bénéficient d'une adaptation de leur poids. On dit alors que l'apprentissage est local car limité à un sous-ensemble des neurones du réseau. [HOW 00]

Une règle d'apprentissage compétitif comporte un mécanisme permettant de mettre les neurones en compétition pour le droit de répondre à un certain sous-ensemble des stimuli d'entrée, de manière à ce qu'un seul neurone de sortie soit actif à la fois.

Ainsi, les neurones individuels peuvent apprendre à se spécialiser sur des sous-ensembles de stimuli similaires pour devenir des détecteurs de caractéristiques. Un neurone vainqueur modifiera ses poids synaptiques en les rapprochant (géométriquement) d'un stimulus d'entrée  $\mathbf{p}$  pour lequel il a battu tous les autres neurones lors de la compétition :

$$\Delta \mathbf{w} = \begin{cases} \eta (\mathbf{p} - \mathbf{w}) & \text{si le neurone est vainqueur.} \\ 0 & \text{autrement.} \end{cases} \quad (2.17)$$

Où  $0 < \eta < 1$  correspond à un taux d'apprentissage.

Un neurone qui ne gagne pas la compétition ne modifiera aucunement ses poids. Il ne sera donc pas affecté par le stimulus en question. Parfois, on définit également un voisinage autour du neurone gagnant et on applique une règle similaire sur les voisins, mais avec un taux d'apprentissage différent :

$$\Delta \mathbf{w} = \begin{cases} \eta_1 (\mathbf{p} - \mathbf{w}) & \text{si le neurone est vainqueur.} \\ \eta_2 (\mathbf{p} - \mathbf{w}) & \text{si le neurone est voisin du vainqueur.} \\ 0 & \text{autrement.} \end{cases} \quad (2.18)$$

Avec  $\eta_1 > \eta_2$



# ANNEXE 3 :

## La règle LMS

### 3.1. Principe de La règle : [PAR 04]

Toujours dans le cadre d'un apprentissage supervisé par correction des erreurs, et comme son nom l'indique, la règle LMS consiste à tenter de minimiser un indice de performance  $F$  basé sur l'erreur quadratique moyenne. Possédant un ensemble d'apprentissage de  $Q$  associations stimulus/cible  $\{(p_q, d_q)\}$ ,  $q = 1, \dots, Q$ , où  $p_q$  représente un vecteur stimulus (entrées) et  $d_q$  un vecteur cible (sorties désirées), à chaque instant  $t$ , on peut propager vers l'avant un stimulus  $p(t)$  à travers le réseau pour obtenir un vecteur de sorties  $a(t)$ . Ceci nous permet de calculer l'erreur  $e(t)$  entre ce que le réseau produit en sortie pour ce stimulus et la cible  $d(t)$  qui lui est associée :

$$e(t) = d(t) - a(t) \quad (3.1)$$

Sachant que tous les neurones d'une même couche sont indépendants les uns des autres, et pour simplifier les équations, nous allons développer la règle LMS pour  $S = 1$ , c'est-à-dire le cas d'un seul neurone. Ensuite, nous pourrons facilement l'étendre au cas général de  $S$  neurones.

Puisque la fonction de transfert est une fonction linéaire nous pouvons écrire la sortie  $a$  du neurone comme suit :

$$a = \mathbf{W}^T \mathbf{p} - b \quad (3.2)$$

Maintenant, nous allons regrouper tous les paramètres libres du neurone (les poids synaptiques et le biais) en un seul vecteur  $\mathbf{x}$  :

$$\mathbf{x} = \begin{bmatrix} \mathbf{W} \\ b \end{bmatrix} \quad (3.3)$$

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

$$\mathbf{y} = \begin{bmatrix} \mathbf{p} \\ -1 \end{bmatrix} \quad (3.4)$$

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

$$a = \mathbf{W}^T \mathbf{p} - b = \mathbf{x}^T \mathbf{y} \quad (3.5)$$

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

$$F(\mathbf{x}) = E[e^2(t)] \quad (3.6)$$

Où  $E[\cdot]$  désigne l'espérance mathématique. Le problème avec cette équation est que l'on ne peut pas facilement calculer cette espérance mathématique puisqu'on ne connaît pas les lois de probabilité de  $\mathbf{x}$ . Une idée plus intéressante, et plus performante en pratique, consiste simplement à estimer l'erreur quadratique moyenne par l'erreur quadratique instantanée pour chaque association d'apprentissage :

$$\hat{F}(\mathbf{x}) = e^2(t) \quad (3.7)$$

Ce qui permet de calculer le gradient :

$$\hat{\nabla} F(\mathbf{x}) = \nabla e^2(t) \quad (3.8)$$

En développant l'équation 3.8 on trouve :

$$\hat{\nabla} F(\mathbf{x}) = \nabla e^2(t) = -2e(t)\mathbf{y}(t) \quad (3.9)$$

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

En appliquant la méthode de la descente du gradient décrite par l'équation 2.14 (cf. Annexe 2) pour modifier les paramètres du neurone dans le sens d'une diminution de  $F$ , nous obtenons la forme générale de la règle LMS, appelée également règle de Widrow-Hoff, écrite sous forme matricielle :

$$\Delta \mathbf{W} = -2 \eta \mathbf{e}(t) \mathbf{p}^T(t) \quad (3.10)$$

$$\Delta \mathbf{b}(t) = -2 \eta \mathbf{e}(t) \quad (3.11)$$

La convergence de la méthode LMS est assurée par la méthode de la descente du gradient qui converge généralement vers un minimum global, à condition de restreindre la valeur du taux d'apprentissage  $\eta$ . En pratique, nous sommes intéressés à fixer  $\eta$  le plus grand possible pour converger le plus rapidement possible (par de grands pas). Mais il existe un seuil à partir duquel un trop grand  $\eta$  peut faire diverger l'algorithme. Pour y remédier, on initialise l'algorithme par un taux d'apprentissage  $0 < \eta < 1/\lambda_{\text{Max}}$  où  $\lambda_{\text{Max}}$  est la plus grande valeur propre de la matrice  $\mathbf{y}\mathbf{y}^T$ .

# ANNEXE 4 :

## La rétropropagation des erreurs

### 4.1. Principe de la règle :

Dans un PMC de M couches nous avons  $\mathbf{a}^k$ ,  $\mathbf{W}^k$ ,  $\mathbf{b}^k$ ,  $f^k$  correspondent respectivement au vecteur sortie, la matrice des poids, le vecteur biais et à la fonction d'activation de la couche k. Par convention nous avons  $\mathbf{a}^0 = \mathbf{p}$ , où  $\mathbf{p}$  est le vecteur d'entrée (stimulus) et la sortie du PMC correspond au vecteur  $\mathbf{a}^M$ . De façon générale nous pouvons écrire l'équation qui décrit la sortie d'une couche k comme étant :

$$\mathbf{a}^k = f^k(\mathbf{W}^k \mathbf{a}^{k-1} - \mathbf{b}^k), \text{ pour } k = 1, \dots, M \quad (4.1)$$

A chaque instant t, on fait propager vers l'avant un stimulus  $\mathbf{p}(t)$  à travers le PMC pour obtenir un vecteur de sorties  $\mathbf{a}(t)$ . Ceci nous permet de calculer l'erreur  $\mathbf{e}(t)$  entre ce que le réseau produit en sortie pour ce stimulus et la cible  $\mathbf{d}(t)$  qui lui est associée. Tout comme pour la règle LMS, L'indice de performance F permet de minimiser l'erreur quadratique moyenne que nous allons approximer par l'erreur instantanée.

$$\hat{F}(\mathbf{x}) = \mathbf{e}^T(t) \mathbf{e}(t) \quad (4.2)$$

La procédure d'optimisation est donc très semblable à celle de la règle LMS. Cependant, il faut faire face à deux difficultés supplémentaires. Premièrement, les fonctions de transfert des neurones ne sont plus nécessairement linéaires. Leurs dérivées partielles ne seront donc plus constantes. Deuxièmement, seules sont connues les sorties désirées (les cibles) pour les neurones de la couche de sortie (couche d'indice M). C'est surtout cette deuxième observation qui va poser problème. [XIN 02]

D'après les résultats de la règle LMS décrite à travers les équations 3.10 et 3.11 (cf. annexe 3):

$$\Delta w_{ij}^k(t) = -\eta \frac{\delta \hat{F}}{\delta w_{ij}^k} \quad (4.3)$$

et

$$\Delta b_i^k(t) = -\eta \frac{\delta \hat{F}}{\delta b_i^k} \quad (4.4)$$

En utilisant les différentes propriétés de la dérivée on peut calculer le deuxième terme des équations 4.3 et 4.4, et qui permet d'écrire :

$$\frac{\delta \hat{F}}{\delta w_{ij}^k} = \frac{\delta \hat{F}}{\delta n_i^k} \times \frac{\delta n_i^k}{\delta w_{ij}^k} \quad (4.5)$$

$$\frac{\delta \hat{F}}{\delta b_i^k} = \frac{\delta \hat{F}}{\delta n_i^k} \times \frac{\delta n_i^k}{\delta b_i^k} \quad (4.6)$$

en rappelant que le niveau d'activation de la couche k peut s'écrire sous la forme :

$$n_i^k = \sum_{j=1}^{S^{(k-1)}} w_{ij}^k a_j^{k-1} - b_i^k \quad (4.7)$$

nous trouvons par conséquent :

$$\frac{\delta n_i^k}{\delta w_{ij}^k} = a_j^{k-1} ; \quad \frac{\delta n_i^k}{\delta b_i^k} = -1 \quad (4.8)$$

Maintenant pour le premier terme du second membre des équations 4.5 et 4.6, nous allons définir une nouvelle entité  $s_i^k$ , qui représente la sensibilité de l'indice de performance F aux changements dans le niveau d'activation  $n_i^k$  du neurone i de la couche k.

$$s_i^k = \frac{\delta \hat{F}}{\delta n_i^k} \quad (4.9)$$

A partir de ce qui précède on peut réécrire les équations 4.3 et 4.4 sous la forme matricielle suivante :

$$\Delta \mathbf{W}^k(t) = -\eta \mathbf{s}^k(t) (\mathbf{a}^{k-1})^T(t) \quad (4.10)$$

$$\Delta \mathbf{b}^k(t) = \eta \mathbf{s}^k(t) \quad (4.11)$$

avec :

$$(\mathbf{s}^k)^T = \left[ \frac{\delta \hat{F}}{\delta n_1^k}, \frac{\delta \hat{F}}{\delta n_2^k}, \dots, \frac{\delta \hat{F}}{\delta n_{S(k)}^k} \right] \quad (4.12)$$

Il reste maintenant à calculer les sensibilités  $s_i^k$ , ce qui requerra une nouvelle fois l'utilisation des propriétés de la dérivée. Dans ce cas, nous obtiendrons une formule de récurrence où la sensibilité des couches en amont (entrées) dépendra de la sensibilité des couches en aval (sorties). C'est de là que provient l'expression «rétropropagation», car le sens de propagation des erreurs est inversé par rapport à celui des informations (stimulus).

Pour dériver la formule de récurrence des sensibilités, nous allons commencer par calculer la matrice suivante :

$$\frac{\delta \mathbf{n}^{k+1}}{\delta \mathbf{n}^k} = \begin{pmatrix} \frac{\delta n_1^{k+1}}{\delta n_1^k} & \frac{\delta n_1^{k+1}}{\delta n_2^k} & \dots & \frac{\delta n_1^{k+1}}{\delta n_{S(k)}^k} \\ \frac{\delta n_2^{k+1}}{\delta n_1^k} & \frac{\delta n_2^{k+1}}{\delta n_2^k} & \dots & \frac{\delta n_2^{k+1}}{\delta n_{S(k)}^k} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\delta n_{S(k+1)}^{k+1}}{\delta n_1^k} & \frac{\delta n_{S(k+1)}^{k+1}}{\delta n_2^k} & \dots & \frac{\delta n_{S(k+1)}^{k+1}}{\delta n_{S(k)}^k} \end{pmatrix} \quad (4.13)$$

Cette matrice énumère toutes les sensibilités des niveaux d'activation d'une couche par rapport à ceux de la couche précédente. Considérons chaque élément (i, j) de cette matrice :

$$\frac{\delta n_i^{k+1}}{\delta n_j^k} = w_{ij}^{k+1} \frac{\delta a_j^k}{\delta n_j^k} = w_{ij}^{k+1} \frac{\delta f(n_j^k)}{\delta n_j^k} = w_{ij}^{k+1} [f^k]'(n_j^k) \quad (4.14)$$

Avec  $[f^k]'$  : la dérivée de la fonction d'activation  $f$  de la  $k^{\text{ème}}$  couche.

Nous pouvons réécrire l'équation 4.14 sous la forme matricielle suivante :

$$\frac{\delta \mathbf{n}^{k+1}}{\delta \mathbf{n}^k} = \mathbf{W}^{k+1} [\mathbf{f}^k]'(\mathbf{n}^k) \quad (4.15)$$

Où :

$$[\mathbf{f}^k]'(\mathbf{n}^k) = \begin{pmatrix} [f^k]'(n_1^k) & 0 & \dots & 0 \\ 0 & [f^k]'(n_2^k) & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & [f^k]'(n_{S(k)}^k) \end{pmatrix} \quad (4.16)$$

Ceci nous permet d'écrire la relation de récurrence pour les sensibilités :

$$\mathbf{s}^k = \frac{\delta \hat{F}}{\delta \mathbf{n}^k} = \begin{bmatrix} \delta \mathbf{n}^{k+1} \\ \delta \mathbf{n}^k \end{bmatrix}^T \times \frac{\delta \hat{F}}{\delta \mathbf{n}^{k+1}} = [\mathbf{f}^k]'(\mathbf{n}^k) (\mathbf{W}^{k+1})^T \mathbf{s}^{k+1} \quad (4.17)$$

Cette équation nous permet de calculer  $\mathbf{s}^1$  à partir de  $\mathbf{s}^2$ , qui lui-même est calculé à partir de  $\mathbf{s}^3$ ,

etc., jusqu'à  $\mathbf{s}^M$ . Ainsi les sensibilités sont rétropropagées de la couche de sortie jusqu'à la couche d'entrée. Il ne nous reste plus qu'à trouver le cas de base,  $\mathbf{s}^M$ , permettant de mettre fin à la récurrence :

$$\begin{aligned}
 s_i^M &= \frac{\delta \hat{F}}{\delta n_i^M} = \frac{\delta (\mathbf{d} - \mathbf{a}^M)^T (\mathbf{d} - \mathbf{a}^M)}{\delta n_i^M} = \frac{\delta}{\delta n_i^M} \left[ \sum_{l=1}^{S(M)} (d_l - a_l^M)^2 \right] \\
 &= -2 (d_i - a_i^M) [\mathbf{f}^M]'(n_i^M)
 \end{aligned} \tag{4.18}$$

Finalement, l'écriture matricielle de la sensibilité de la couche de sortie M :

$$\mathbf{s}^M = -2 [\mathbf{f}^M]'(\mathbf{n}^M) (\mathbf{d} - \mathbf{a}^M) \tag{4.19}$$

**ANNEXE 5 :***Tableaux des données*

années	gasoil	essence
1980	1974127	1212970
1981	2161911	1326221
1982	2284736	1393504
1983	2635585	1528789
1984	2906280	1715085
1985	3127540	1784843
1986	3236904	1897157
1987	3337156	1907341
1988	3246494	1962230
1989	3435207	2093969
1990	3390503	2169015
1991	3673728	2267384
1992	3577702	2217840
1993	3446211	2333354
1994	3225275	2209085
1995	3131433	2064824
1996	3097795	2093061
1997	3023384	2003062
1998	3351496	1978285
1999	3434132	1923059
2000	3697428	1888647
2001	3929919	1840213
2002	4560486	1878819
2003	4833706,937	1775146,95
2004	5124643,577	1657399,03

**Tableau 5.1 : Evolution annuelle de la consommation des carburants [SONATACH].**

années	Population (en millier)	population active (en millier)	population urbaine (en millier)	PIB industriel MDA	Dépenses ménage (MDA)	parc auto	dépenses gouvernement (MDA)
1980	18666000	4853160	8119710	88213.56919	17656.25	1213089	24699.9936
1981	19262000	5046644	8590852	100943.4186	19583.33	1380558	28899.9792
1982	19883000	5268995	9046765	112778.6745	20174.29	1524690	33599.9934
1983	20522000	5479374	9522208	126837.6302	21503.13	1678068	37900.0128
1984	21185000	5698765	9999320	142001.1556	27168.67	1869630	42900.0108
1985	21863000	5946736	10494240	154407.8061	30039.76	2002851	48700.0073
1986	22512000	6168288	11008368	159494.9832	36808.51	2111239	56899.986
1987	23139000	6386364	11546361	167138.5942	38144.33	2226095	62899.9895
1988	23783000	6611674	12081764	186309.8201	35093.06	2306727	70299.9819
1989	24409000	6810111	12619453	219380.5934	34375.82	2396115	76199.9954
1990	25022000	7031182	13161572	292669.964	35256.7	2480450	89300.0192
1991	25643000	7359541	13744648	451362.0829	21911.21	2541520	124299.9601
1992	26271000	7697403	14343966	551735.72	24848.9	2587731	167900.0232
1993	26894000	8014412	14926170	613315.4735	27446.44	2636553	202086.5445
1994	27496000	8386280	15535240	757453.8875	23639.48	2678364	266600.0966
1995	28060000	8726660	16106440	1010528.894	23025.6	2727147	333700.0688
1996	28566000	9055422	16711110	1276568.441	24438.36	2773356	420899.9325
1997	29045000	9410580	17310820	1416137.875	24432.51	2809504	462299.8054
1998	29507000	9766817	17910749	1422508.593	25961.87	2845491	503499.8958
1999	29965000	10128170	18488405	1645499.89058791	24545.59	2892651	540899.8896
2000	30416000	10493520	19040416	2128601.22908558	22763.81	2914415	560201.573
2001	30879000	10869408	19639044	2187406.036914	23855.47	2938454	630540.6822

**Tableau 5.2 : Les évolutions des différents facteurs influents la consommation des carburants.**

[APRUE + ONS+ CNIS + [www.afdb.org](http://www.afdb.org)]



**ANNEXE 6 :***Classement de facteurs par la méthode de Gram Schmidt***6.1 Le modèle gasoil :**

années	SORTIE	ENTREES							
	gasoil	essence	population	population active	population urbaine	PIB industriel	dépenses ménage	parc auto	dépenses gouvernement
1980	-1177006,91	-687481,727	-6469818,18	-2656363	-5514272,77	-603208,9	-8828,89045	-1129397,27	-192810,39
1981	-989222,909	-574230,727	-5873818,18	-2462879	-5043130,77	-590479,051	-6901,81045	-961928,273	-188610,404
1982	-866397,909	-506947,727	-5252818,18	-2240528	-4587217,77	-578643,795	-6310,85045	-817796,273	-183910,39
1983	-515548,909	-371662,727	-4613818,18	-2030149	-4111774,77	-564584,839	-4982,01045	-664418,273	-179610,371
1984	-244853,909	-185366,727	-3950818,18	-1810758	-3634662,77	-549421,314	683,529545	-472856,273	-174610,373
1985	-23593,9091	-115608,727	-3272818,18	-1562787	-3139742,77	-537014,663	3554,61955	-339635,273	-168810,376
1986	85770,0909	-3294,72727	-2623818,18	-1341235	-2625614,77	-531927,486	10323,3695	-231247,273	-160610,398
1987	186022,091	6889,27273	-1996818,18	-1123159	-2087621,77	-524283,875	11659,1895	-116391,273	-154610,394
1988	95360,0909	61778,2727	-1352818,18	-897849	-1552218,77	-505112,649	8607,91955	-35759,2727	-147210,402
1989	284073,091	193517,273	-726818,182	-699412	-1014529,77	-472041,876	7890,67955	53628,7273	-141310,388
1990	239369,091	268563,273	-113818,182	-478341	-472410,773	-398752,506	8771,55955	137963,727	-128210,364
1991	522594,091	366932,273	507181,818	-149982	110665,227	-240060,387	-4573,93045	199033,727	-93210,4236
1992	426568,091	317388,273	1135181,82	187880	709983,227	-139686,75	-1636,24045	245244,727	-49610,3605
1993	295077,091	432902,273	1758181,82	504889	1292187,23	-78106,9961	961,299545	294066,727	-15423,8392
1994	74141,0909	308633,273	2360181,82	876757	1901257,23	66031,4179	-2845,66045	335877,727	49089,7129
1995	-19700,9091	164372,273	2924181,82	1217137	2472457,23	319106,424	-3459,54045	384660,727	116189,685
1996	-53338,9091	192609,273	3430181,82	1545899	3077127,23	585145,971	-2046,78045	430869,727	203389,549
1997	-127749,909	102610,273	3909181,82	1901057	3676837,23	724715,405	-2052,63045	467017,727	244789,422
1998	200362,091	77833,2727	4371181,82	2257294	4276766,23	731086,123	-523,270455	503004,727	285989,512
1999	282998,091	22607,2727	4829181,82	2618647	4854422,23	954077,421	-1939,55045	550164,727	323389,506
2000	546294,091	-11804,7273	5280181,82	2983997	5406433,23	1437178,76	-3721,33045	571928,727	342691,189
2001	778785,091	-60238,7273	5743181,82	3359885	6005061,23	1495983,57	-2629,67045	595967,727	413030,299

**Tableau 6.1 : données centrées pour le modèle Gasoil.**

1<sup>ère</sup> itération :

Descripteurs	Cos <sup>2</sup>
parc auto	0,7326
Essence	0,6609
Population	0,5766
Population urbaine	0,5264
Population active	0,4844
PIB industriel	0,2845
dépenses gouvernement	0,2678
dépenses ménages	0,1744

Tableau 6.2 : classement des descripteurs selon la valeur du cos<sup>2</sup>.2<sup>ème</sup> itération :

années	ENTREES							
	SORTIE	gasoil	essence	population	population active	population urbaine	PIB industriel	dépenses ménages
1980	-262050	-155170	1580100	1039200	1816500	539600	-6707	162880
1981	-209940	-120850	982400	684700	1200600	382870	-5095	114340
1982	-203880	-121500	576100	435400	721000	248860	-4775	73640
1983	22720	-58510	121900	143900	200900	107720	-3734	29640
1984	138220	37500	-580500	-263500	-565400	-70950	1572	-25690
1985	251550	44470	-852000	-451500	-935200	-193350	4193	-61850
1986	273110	105700	-975600	-584600	-1124600	-297930	10758	-87780
1987	280310	61750	-1167200	-742300	-1332100	-406510	11878	-117950
1988	124330	78630	-1097900	-780800	-1320100	-468930	8675	-135950
1989	240630	168240	-1109100	-874900	-1362600	-526310	7790	-158200
1990	127600	203540	-1097200	-929800	-1367900	-538350	8512	-171660
1991	361350	273120	-911500	-801200	-1181200	-441460	-4948	-155890
1992	227890	201800	-612800	-614600	-881900	-387840	-2097	-126850
1993	56840	294300	-337800	-457300	-616600	-375670	409	-108040
1994	-197960	150330	-33800	-222300	-278900	-273830	-3477	-56690
1995	-331330	-16930	182500	-41500	-24300	-70120	-4182	-4950
1996	-402400	-10470	359100	136000	280400	149160	-2856	67690
1997	-506090	-117510	580500	372900	645500	252150	-2930	97710
1998	-207140	-159240	786000	611400	1011800	222110	-1468	127570
1999	-162710	-236700	907800	818400	1283400	397380	-2973	150120
2000	82960	-281370	1203700	1112600	1694100	858460	-4796	162570
2001	295970	-341130	1495400	1409800	2136700	892940	-3749	225340

Tableau 6.3 : Données projetées par rapport au sous-espace orthogonal au parc automobile.

Descripteurs	Cos <sup>2</sup>
population	0,3423
dépenses ménages	0,2902
dépenses gouvernement	0,2807
population urbaine	0,2622
population active	0,2223
PIB industriel	0,1560
essence	0,1521

Tableau 6.4 : classement des descripteurs selon le Cos<sup>2</sup>.3<sup>ème</sup> itération :

années	SORTIE		ENTREES				
	gasoil	Essence	population active	population urbaine	PIB industriel	dépenses ménages	dépenses gouvernement
1980	-2900	103830	-170170	-159740	-163900	1126	-44029
1981	-48820	40180	-67200	-28090	-54520	-225	-14302
1982	-109390	-27070	-5530	470	-7630	-1919.1	-1798
1983	42710	-38530	50600	48440	53450	-3129.7	13678
1984	43010	-57650	180800	160630	187500	-1305.7	50324
1985	111810	-95180	200600	130400	185980	-30.6	49716
1986	113100	-54210	162100	95590	136430	5921.7	39971
1987	88880	-129570	151040	127720	113150	6091.9	34891
1988	-55740	-101330	59500	53050	19880	3232.4	7816
1989	58730	-13550	-26030	24560	-32510	2291.9	-12967
1990	-52350	23700	-90030	4370	-49850	3072.9	-27986
1991	211850	123720	-103560	-41180	-35640	-9466.5	-36532
1992	127380	101360	-145580	-115470	-115010	-5134.8	-46606
1993	1440	238930	-198760	-194110	-225270	-1265.6	-63806
1994	-203500	144790	-196430	-236630	-258780	-3644.6	-52264
1995	-301400	12980	-181180	-252550	-151370	-3277.3	-28848
1996	-343500	48390	-138850	-168730	-10720	-1075.9	20667
1997	-410880	-22360	-71400	-80530	-6300	-52.3	21696
1998	-78230	-30410	9820	28740	-127840	2428.4	24646
1999	-13820	-87900	123590	148010	-6790	1527.2	31247
2000	280380	-84070	191320	188620	322540	1171	4950
2001	541230	-96020	265260	266390	227150	3664.1	29523

Tableau 6.5 : Données projetées par rapport au sous-espace orthogonal à la population.

Descripteurs	Cos <sup>2</sup>
population urbaine	0,4609
population active	0,3673
PIB industriel	0,3130
essence	0,0591
dépenses ménages	0,0261
dépenses gouvernement	0,0201

Tableau 6.6 : classement des descripteurs selon le Cos<sup>2</sup>.4<sup>ème</sup> itération :

années	SORTIE	ENTREES				
	Gasoil	essence	population active	PIB industriel	dépenses ménages	dépenses gouvernement
1980	154310	19400	-13000	-21680	3188.3	-15252
1981	-21170	25330	-39562	-29510	137.7	-9242
1982	-109850	-26820	-5992	-8050	-1925.2	-1883
1983	-4960	-12930	2939	10320	-3755.1	4952
1984	-115080	27250	22754	44490	-3379.5	21387
1985	-16530	-26260	72298	69880	-1714.1	26225
1986	19020	-3690	68048	51320	4687.6	22751
1987	-36820	-62060	25375	-560	4443	11882
1988	-107950	-73290	7304	-27350	2547.5	-1741
1989	34560	-570	-50195	-54380	1974.8	-17391
1990	-56650	26010	-94330	-53740	3016.5	-28773
1991	252380	101950	-63043	1020	-8934.8	-29113
1992	241020	40330	-31968	-12200	-3644	-25804
1993	192480	136340	-7773	-52450	1240.4	-28837
1994	29380	19720	36393	-48100	-589.6	-9635
1995	-52850	-120500	67307	73480	-16.8	16649
1996	-177440	-40790	27165	139500	1102.5	51064
1997	-331620	-64920	7834	65400	987.4	36203
1998	-106510	-15220	-18458	-153430	2057.4	19469
1999	-159490	-9670	-22039	-138570	-383.7	4583
2000	94750	15620	5735	154610	-1264.2	-29030
2001	279060	44780	3156	-10020	224.9	-18467

Tableau 6.7: Données projetées par rapport au sous-espace orthogonal à la population urbaine.

Descripteurs	Cos <sup>2</sup>
dépenses gouvernement	0,5450
essence	0,4765
dépenses ménages	0,0858
population active	0,0470
PIB industriel	0,0051

Tableau 6.8 : Classement des descripteurs selon le Cos<sup>2</sup>.5<sup>ème</sup> itération :

années	SORTIE	ENTREES			
	gasoil	essence	population active	PIB industriel	dépenses ménages
1980	80420	-4833	4289	-5490	3637.1
1981	-65940	10646	-29086	-19700	409.7
1982	-118970	-29812	-3858	-6050	-1869.8
1983	19030	-5062	-2674	5060	-3900.8
1984	-11470	61231	-1490	21790	-4008.9
1985	110520	15408	42570	42050	-2485.9
1986	129240	32458	42258	27180	4018.1
1987	20740	-43181	11906	-13170	4093.3
1988	-116380	-76056	9278	-25500	2598.7
1989	-49690	-28202	-30481	-35920	2486.6
1990	-196040	-19706	-61714	-23210	3863.2
1991	111340	55694	-30042	31910	-8078.1
1992	116010	-669	-2718	15180	-2884.6
1993	52780	90522	24916	-21850	2089
1994	-17300	4411	47315	-37880	-306.1
1995	27810	-94047	48434	55810	-506.7
1996	69940	40343	-30719	85310	-400.2
1997	-156240	-7399	-33204	26980	-78
1998	-12190	15713	-40527	-174090	1484.5
1999	-137290	-2388	-27234	-143430	-518.6
2000	-45890	-30504	38642	185420	-409.9
2001	189600	15439	24089	9580	768.3

Tableau 6.9 : Données projetées par rapport au sous-espace orthogonal aux dépenses gouvernement.

Descripteurs	Cos <sup>2</sup>
population active	0,2309
essence	0,1628
PIB industriel	0,0735
dépenses ménages	0,0396

Tableau 6.10: Classement des descripteurs selon le Cos<sup>2</sup>.

6<sup>ème</sup> itération :

années	SORTIE	ENTREES		
	gasoil	essence	PIB industriel	dépense ménages
1980	73840	-4144	-9340	3620.6
1981	-21290	5976	6400	521.7
1982	-113050	-30431	-2590	-1854.9
1983	23140	-5491	7460	-3890.5
1984	-9180	60992	23130	-4003.2
1985	45170	22244	3850	-2649.8
1986	64360	39243	-10740	3855.4
1987	2460	-41269	-23850	4047.5
1988	-130620	-74566	-33830	2563
1989	-2900	-33096	-8570	2603.9
1990	-101300	-29616	32170	4100.8
1991	157460	50870	58870	-7962.4
1992	120180	-1105	17620	-2874.1
1993	14530	94523	-44210	1993.1
1994	-89940	12008	-80340	-488.3
1995	-46550	-86270	12350	-693.2
1996	117100	35410	112880	-281.9
1997	-105260	-12731	56780	49.8
1998	50030	9206	-137720	1640.5
1999	-95480	-6761	-118990	-413.8
2000	-105210	-24299	150740	-558.7
2001	152620	19307	-12040	675.6

Tableau 6.11 : Données projetées par rapport au sous-espace orthogonal à la population active.

Descripteurs	Cos <sup>2</sup>
essence	0,2819
dépenses ménages	0,0622
PIB industriel	0,0086

Tableau 6.12 : classement des descripteurs selon le Cos<sup>2</sup>.

7<sup>ème</sup> itération :

années	SORTIE	ENTREES	
	gasoil	PIB industriel	dépenses ménages
1980	78560	-9370	3535
1981	-28100	6450	645.2
1982	-78370	-2830	-2483.7
1983	29400	7420	-4004
1984	-78680	23600	-2743
1985	19820	4020	-2190.2
1986	19640	-10440	4666.2
1987	49490	-24170	3194.8
1988	-45650	-34410	1022.3
1989	34810	-8830	1920.1
1990	-67550	31940	3488.9
1991	99490	59260	-6911.3
1992	121440	17610	-2896.9
1993	-93180	-43480	3946.1
1994	-103620	-80250	-240.2
1995	51750	11680	-2475.7
1996	76750	113150	449.7
1997	-90750	56680	-213.2
1998	39540	-137650	1830.7
1999	-87780	-119040	-553.5
2000	-77520	150550	-1060.8
2001	130620	-11890	1074.5

Tableau 6.13 : Données projetées par rapport au sous-espace orthogonal à l'essence.

Descripteurs	Cos <sup>2</sup>
dépenses ménages	0,0145
PIB industriel	0,0127

Tableau 6.14 : classement des descripteurs selon le Cos<sup>2</sup>.

Le classement final :

1. Parc auto.
2. Population.
3. Population urbaine.
4. Dépenses du gouvernement à la consommation finale.
5. Population active.
6. Essence.
7. Dépense ménages.
8. PIB industriel.

6.2. Le modèle essence :

années	SORTIE	ENTREES							
	essence	gasoil	population	population active	population urbaine	PIB industriel	dépenses ménage	parc auto	dépenses gouvernement
1980	-687481,727	-1177006,91	-6469818,18	-2656363	-5514272,77	-603208,9	-8828,89045	-1129397,27	-192810,39
1981	-574230,727	-989222,909	-5873818,18	-2462879	-5043130,77	-590479,051	-6901,81045	-961928,273	-188610,404
1982	-506947,727	-866397,909	-5252818,18	-2240528	-4587217,77	-578643,795	-6310,85045	-817796,273	-183910,39
1983	-371662,727	-515548,909	-4613818,18	-2030149	-4111774,77	-564584,839	-4982,01045	-664418,273	-179610,371
1984	-185366,727	-244853,909	-3950818,18	-1810758	-3634662,77	-549421,314	683,529545	-472856,273	-174610,373
1985	-115608,727	-23593,9091	-3272818,18	-1562787	-3139742,77	-537014,663	3554,61955	-339635,273	-168810,376
1986	-3294,72727	85770,0909	-2623818,18	-1341235	-2625614,77	-531927,486	10323,3695	-231247,273	-160610,398
1987	6889,27273	186022,091	-1996818,18	-1123159	-2087621,77	-524283,875	11659,1895	-116391,273	-154610,394
1988	61778,2727	95360,0909	-1352818,18	-897849	-1552218,77	-505112,649	8607,91955	-35759,2727	-147210,402
1989	193517,273	284073,091	-726818,182	-699412	-1014529,77	-472041,876	7890,67955	53628,7273	-141310,388
1990	268563,273	239369,091	-113818,182	-478341	-472410,773	-398752,506	8771,55955	137963,727	-128210,364
1991	366932,273	522594,091	507181,818	-149982	110665,227	-240060,387	-4573,93045	199033,727	-93210,4236
1992	317388,273	426568,091	1135181,82	187880	709983,227	-139686,75	-1636,24045	245244,727	-49610,3605
1993	432902,273	295077,091	1758181,82	504889	1292187,23	-78106,9961	961,299545	294066,727	-15423,8392
1994	308633,273	74141,0909	2360181,82	876757	1901257,23	66031,4179	-2845,66045	335877,727	49089,7129
1995	164372,273	-19700,9091	2924181,82	1217137	2472457,23	319106,424	-3459,54045	384660,727	116189,685
1996	192609,273	-53338,9091	3430181,82	1545899	3077127,23	585145,971	-2046,78045	430869,727	203389,549
1997	102610,273	-127749,909	3909181,82	1901057	3676837,23	724715,405	-2052,63045	467017,727	244789,422
1998	77833,2727	200362,091	4371181,82	2257294	4276766,23	731086,123	-523,270455	503004,727	285989,512
1999	22607,2727	282998,091	4829181,82	2618647	4854422,23	954077,421	-1939,55045	550164,727	323389,506
2000	-11804,7273	546294,091	5280181,82	2983997	5406433,23	1437178,76	-3721,33045	571928,727	342691,189
2001	-60238,7273	778785,091	5743181,82	3359885	6005061,23	1495983,57	-2629,67045	595967,727	413030,299

Tableau 6.15 : Données centrées pour le modèle Essence.



1<sup>ère</sup> itération :

Descripteurs	Cos <sup>2</sup>
gasoil	0,6609
population	0,4512
Population active	0,2947
Population urbaine	0,3627
PIB industriel	0,0806
dépenses ménages	0,1617
parc auto	0,6601
dépenses gouvernement	0,1008

Tableau 6.16 : classement des descripteurs selon le Cos<sup>2</sup>.2<sup>ème</sup> itération :

années	SORTIE	ENTREES						
	Essence *10 <sup>5</sup>	population 10 <sup>6</sup>	population active*10 <sup>6</sup>	population urbaine*10 <sup>6</sup>	PIB industriel *10 <sup>6</sup>	dépenses ménages *10 <sup>4</sup>	parc auto*10 <sup>5</sup>	dépenses gouvernement *10 <sup>5</sup>
1980	-101020	0459200	0412200	0669300	0.2637	-0.2989	-0.6509	0.6126
1981	-081330	-0050300	0116100	0153900	0.1381	-0.1994	-0.6742	0.2492
1982	-075250	-0152300	0018200	-0035500	0.0595	-0.2012	-0.3436	0.0311
1983	-114780	-1578800	-0686100	-1403300	-0.1849	-0.2424	-1.9823	-0.6832
1984	-063360	-2509400	-1172400	-2348300	-0.3691	0.1898	-2.5145	-1.2176
1985	-103850	-3133900	-1501300	-3015800	-0.5196	0.3672	-3.1830	-1.6372
1986	-046030	-3128700	-1564800	-3076200	-0.5951	0.9898	-3.0880	-1.7912
1987	-085800	-3091900	-1608100	-3064900	-0.6613	1.0736	-2.8460	-1.9477
1988	014260	-1914200	-1146500	-2053200	-0.5753	0.8135	-1.2199	-1.6779
1989	051970	-2399200	-1440000	-2506900	-0.6813	0.6481	-2.0324	-2.0263
1990	149290	-1523000	-1102400	-1730000	-0.5750	0.7584	-0.7849	-1.7988
1991	106540	-2569300	-1512400	-2634800	-0.6250	-0.7167	-2.7352	-2.0602
1992	104840	-1376000	-0924200	-1531000	-0.4539	-0.3753	-1.4048	-1.4169
1993	285880	0021100	-0264400	-0258000	-0.2954	-0.0503	0.2724	-0.7912
1994	271690	1923700	0683500	1511700	0.0114	-0.3214	2.6884	0.3309
1995	174190	3040200	1268500	2576000	0.3336	-0.3362	4.0248	1.2044
1996	219190	3744200	1685000	3357300	0.6244	-0.1782	4.7910	2.1490
1997	166260	4661200	2234100	4348000	0.8188	-0.1419	5.8254	2.7237
1998	-022000	3191700	1734900	3224100	0.5835	-0.1517	3.2183	2.4274
1999	-118400	3163200	1880900	3367700	0.7456	-0.3344	2.9426	2.6230
2000	-284010	2064200	1559800	2536400	1.0348	-0.6432	0.7794	2.2477
2001	-448280	1158500	1329500	1913600	0.9224	-0.6494	-1.0825	2.4492

Tableau 6.17 : Données projetées par rapport au sous-espace orthogonal au gasoil.

Descripteurs	Cos <sup>2</sup>
parc auto	0,15
PIB industriel	0,0925
dépenses gouvernement	0,0429
population	0,0206
Dépenses ménages	0,014
population active	0,003
population urbaine	0,00095955

Tableau 6.18 : classement des descripteurs selon le Cos<sup>2</sup>.2<sup>ème</sup> itération :

années	SORTIE	ENTREES					
	Eseence *10 <sup>^5</sup>	PIB industriel *10 <sup>^6</sup>	Dépenses gouvernement *10 <sup>^5</sup>	Population *10 <sup>^6</sup>	dépenses ménages *10 <sup>^3</sup>	population active *10 <sup>^6</sup>	population urbaine *10 <sup>^6</sup>
1980	-0.8455	0.3648	0.9535	1.0332	-3.5100	0.6945	1.2132
1981	-0.6427	0.2428	0.6023	0.5442	-2.5337	0.4086	0.7173
1982	-0.6656	0.1128	0.2110	0.1507	-2.2870	0.1672	0.2516
1983	-0.6463	0.1229	0.3550	0.1693	-4.0108	0.1738	0.2531
1984	0.0026	0.0213	0.0993	-0.2920	-0.1148	-0.0817	-0.2472
1985	-0.2332	-0.0254	0.0298	-0.3270	1.1241	-0.1206	-0.3561
1986	0.3210	-0.1157	-0.1740	-0.4056	7.4262	-0.2253	-0.4959
1987	-0.1380	-0.2195	-0.4572	-0.5822	8.4579	-0.3736	-0.6868
1988	0.4512	-0.3859	-1.0390	-0.8384	7.1585	-0.6173	-1.0338
1989	1.0339	-0.3658	-0.9619	-0.6069	4.8541	-0.5584	-0.8086
1990	1.6915	-0.4531	-1.3877	-0.8308	6.9557	-0.7619	-1.0741
1991	1.7574	-0.2004	-0.6277	-0.1573	-9.3564	-0.3259	-0.3493
1992	1.4038	-0.2358	-0.6812	-0.1372	-4.8775	-0.3148	-0.3571
1993	2.7899	-0.3377	-0.9339	-0.2191	-0.2850	-0.3826	-0.4856
1994	2.0367	-0.4060	-1.0770	-0.4470	-1.0620	-0.4827	-0.7347
1995	0.7236	-0.2913	-0.9034	-0.5090	-0.1403	-0.4774	-0.7871
1996	0.9798	-0.1194	-0.3601	-0.4807	2.0530	-0.3932	-0.6461
1997	0.1888	-0.0856	-0.3271	-0.4759	3.2440	-0.2928	-0.5197
1998	-1.0342	0.0839	0.7419	0.3537	1.0591	0.3389	0.5349
1999	-1.9285	0.2888	1.0819	0.5683	-0.9885	0.6045	0.9089
2000	-3.0373	0.9138	1.8395	1.3769	-5.8081	1.2217	1.8851
2001	-4.2089	1.0905	3.0161	2.1131	-7.3605	1.7991	2.8181

Tableau 6.19 : Données projetées par rapport au sous-espace orthogonal au parc auto.

Descripteurs	Cos <sup>2</sup>
Dépenses gouvernement	0,8824
PIB industriel	0,8533
population active	0,8252
population urbaine	0,768
Population	0,6849
Dépenses ménages	0,1067

Tableau 6.20 : classement des descripteurs selon le Cos<sup>2</sup>.3<sup>ème</sup> itération :

années	SORTIE	ENTREES				
	essence *10 <sup>5</sup>	PIB Industriel *10 <sup>5</sup>	population active*10 <sup>5</sup>	population urbaine*10 <sup>5</sup>	population *10 <sup>5</sup>	dépenses ménages*10 <sup>4</sup>
1980	0.5360	0.1245	1.3352	3.4634	4.0091	-0.1187
1981	0.2299	0.2023	0.5425	1.6973	1.4480	-0.1066
1982	-0.3599	0.3483	0.4306	0.5977	0.1078	-0.1773
1983	-0.1320	-0.0828	-0.3506	-0.6964	-0.6611	-0.3146
1984	0.1465	-0.1539	-1.4012	-3.3748	-3.5785	0.0127
1985	-0.1900	-0.3641	-1.3813	-3.8319	-3.4676	0.1197
1986	0.0689	-0.5140	-1.2293	-3.3771	-2.9022	0.7002
1987	-0.8004	-0.5055	-1.0461	-2.7114	-2.7902	0.7344
1988	-1.0541	-0.0196	-0.0602	-0.8921	-1.4941	0.4627
1989	-0.3597	-0.1035	0.0752	0.6590	0.3096	0.2510
1990	-0.3190	0.5970	0.5453	1.8750	0.8942	0.3574
1991	0.8480	0.3155	0.4340	2.2136	2.5894	-1.0886
1992	0.4169	0.1592	0.8597	2.6220	3.1452	-0.6537
1993	1.4368	0.0741	1.6685	3.6344	4.0019	-0.2561
1994	0.4763	-0.0801	1.5094	2.4444	2.6719	-0.3686
1995	-0.5853	0.4253	0.5410	0.3421	0.9007	-0.2342
1996	0.4581	0.1367	-1.8134	-3.1872	-2.4191	0.1176
1997	-0.2851	0.3527	-1.0036	-2.2232	-2.5899	0.2447
1998	0.0407	-1.9026	-0.9759	-1.3959	-1.3827	0.2867
1999	-0.3610	-1.1100	-0.3202	-0.7469	-1.4914	0.1648
2000	-0.3722	2.3405	1.3946	2.1275	1.5708	-0.1326
2001	0.1609	-0.2404	0.2462	0.7606	1.1305	-0.0012

Tableau 6.21 : Données projetées par rapport au sous-espace orthogonal aux dépenses du gouvernement.

Descripteur	Cos <sup>2</sup>
dépenses ménages	0,3331
population	0,2953
population urbaine	0,1903
population active	0,1046
PIB industriel	0,00044543

Tableau 6.22 : classement des descripteurs selon le Cos<sup>2</sup>.4<sup>ème</sup> itération :

années	SORTIE	ENTREES			
	Essence *10 <sup>^5</sup>	population *10 <sup>^5</sup>	population urbaine*10 <sup>^5</sup>	population active*10 <sup>^5</sup>	PIB industriel *10 <sup>^5</sup>
1980	0.4439	3.5522	3.0539	1.1759	0.0553
1981	0.1472	1.0377	1.3296	0.3995	0.1401
1982	-0.4975	-0.5747	-0.0139	0.1927	0.2449
1983	-0.3762	-1.8721	-1.7817	-0.7727	-0.2663
1984	0.1564	-3.5296	-3.3310	-1.3842	-0.1465
1985	-0.0971	-3.0068	-3.4190	-1.2207	-0.2943
1986	0.6124	-0.2069	-0.9617	-0.2898	-0.1056
1987	-0.2303	0.0367	-0.1780	-0.0607	-0.0771
1988	-0.6949	0.2870	0.7040	0.5607	0.2503
1989	-0.1649	1.2758	1.5249	0.4120	0.0429
1990	-0.0416	2.2699	3.1079	1.0249	0.8055
1991	0.0030	-1.6009	-1.5417	-1.0267	-0.3195
1992	-0.0905	0.6289	0.3670	-0.0174	-0.2221
1993	1.2380	3.0161	2.7510	1.3249	-0.0753
1994	0.1902	1.2531	1.1729	1.0148	-0.2951
1995	-0.7671	-0.0008	-0.4658	0.2267	0.2887
1996	0.5494	-1.9664	-2.7815	-1.6556	0.2053
1997	-0.0952	-1.6480	-1.3791	-0.6753	0.4954
1998	0.2632	-0.2791	-0.4069	-0.5912	-1.7354
1999	-0.2331	-0.8570	-0.1784	-0.0991	-1.0139
2000	-0.4751	1.0604	1.6701	1.2167	2.2632
2001	0.1600	1.1259	0.7565	0.2446	-0.2411

Tableau 6.23 : Données projetées par rapport au sous-espace orthogonal aux dépenses ménages.

Descripteurs	Cos <sup>2</sup>
PIB industriel	0,0715
population	0,0692
population urbaine	0,0178
population active	0,000335

Tableau 6.24 : classement des descripteurs selon le Cos<sup>2</sup>.5<sup>ème</sup> itération :

années	SORTIE	ENTREES		
	Essence *10 <sup>5</sup>	population *10 <sup>5</sup>	population urbaine*10 <sup>5</sup>	population active*10 <sup>5</sup>
1980	0.4534	3.5210	3.0124	1.1499
1981	0.1712	0.9586	1.2243	0.3335
1982	-0.4556	-0.7129	-0.1979	0.0773
1983	-0.4218	-1.7218	-1.5816	-0.6473
1984	0.1313	-3.4469	-3.2209	-1.3152
1985	-0.1475	-2.8407	-3.1979	-1.0821
1986	0.5943	-0.1473	-0.8824	-0.2401
1987	-0.2435	0.0802	-0.1201	-0.0244
1988	-0.6521	0.1457	0.5160	0.4428
1989	-0.1576	1.2516	1.4927	0.3918
1990	0.0962	1.8152	2.5027	0.6455
1991	-0.0517	-1.4205	-1.3017	-0.8762
1992	-0.1285	0.7543	0.5339	0.0872
1993	1.2251	3.0586	2.8076	1.3604
1994	0.1397	1.4197	1.3946	1.1538
1995	-0.7177	-0.1638	-0.6827	0.0907
1996	0.5845	-2.0823	-2.9357	-1.7523
1997	-0.0104	-1.9277	-1.7513	-0.9086
1998	-0.0337	0.7005	0.8969	0.2262
1999	-0.4066	-0.2847	0.5833	0.3785
2000	-0.0879	-0.2172	-0.0302	0.1507
2001	0.1187	1.2620	0.9376	0.3582

Tableau 6.25 : Données projetées par rapport au sous-espace orthogonal au PIB industriel.

Descripteurs	Cos <sup>2</sup>
population	0,118
population urbaine	0,0514
population active	0,0186

Tableau 6.26 : classement des descripteurs selon le Cos<sup>2</sup>.

6<sup>ème</sup> itération :

années	SORTIE	ENTREES	
	Essence *10 <sup>4</sup>	population urbaine*10 <sup>4</sup>	population active*10 <sup>4</sup>
1980	1.5115	-4.8782	-3.3334
1981	0.8891	2.7136	-0.7032
1982	-3.9440	5.1079	3.7761
1983	-2.7400	1.3004	0.7802
1984	4.2719	2.0566	1.3682
1985	0.9635	-3.7397	1.1456
1986	6.0694	-7.3597	-1.7805
1987	-2.5038	-1.9983	-0.5818
1988	-6.6461	3.7116	3.8142
1989	-2.6504	2.4849	-1.3544
1990	-0.5962	6.9821	-1.1916
1991	0.7024	1.1042	-2.7781
1992	-1.9325	-2.1595	-2.3055
1993	9.6255	-2.3295	0.7195
1994	0.1783	-0.1672	5.5574
1995	-7.0364	-5.1987	1.5970
1996	7.6325	-8.6569	-8.7512
1997	1.5508	1.6502	-0.9655
1998	-0.9383	2.0053	-0.6889
1999	-3.8216	8.6632	4.9843
2000	-0.6926	1.8572	2.4220
2001	0.1037	-3.1695	-1.7342

Tableau 6.27 : Données projetées par rapport au sous-espace orthogonal à la population.

Descripteurs	Cos <sup>2</sup>
pop active	0.2413
pop urbaine	0.2154

Tableau 6.28 : classement des descripteurs selon le Cos<sup>2</sup>.

Le classement final :

1. Gasoil.
2. Parc auto.
3. Dépenses gouvernement à la consommation finale.
4. Dépense ménages.
5. PIB industriel.
6. Population.
7. Population active.
8. Population urbaine.

**ANNEXE 7 :***Le test de Fisher***7.1. Le modèle gasoil :**

années	d
1980	-1177006.90909091
1981	-989222.90909091
1982	-866397.90909091
1983	-515548.90909091
1984	-244853.90909091
1985	-23593.9090909101
1986	85770.0909090899
1987	186022.09090909
1988	95360.0909090899
1989	284073.09090909
1990	239369.09090909
1991	522594.09090909
1992	426568.09090909
1993	295077.09090909
1994	74141.0909090899
1995	-19700.9090909101
1996	-53338.9090909101
1997	-127749.90909091
1998	200362.09090909
1999	282998.09090909
2000	546294.09090909
2001	778785.09090909

**Tableau 7.1 : l'évolution observée de la consommation annuelle du gasoil (centrée).***Itération 0 : Le modèle complet.*

Descripteurs	$\beta$
essence	1.0883
population	-1.6355
population active	1.6283
population urbaine	0.7998
PIB industriel	0.1011
dépenses ménages	-2.5183
parc auto	2.9110
dépenses gouvernement	-4.9954

**Tableau 7.2 : les coefficients MCO du modèle complet.**

$d^{(\text{complet})} * 10^6$	$(d - d^{(\text{complet})}) * 10^5$
-1.2655	0.8845
-0.9622	-0.2706
-0.7821	-0.8429
-0.5341	0.1854
-0.1569	-0.8791
-0.0375	0.1386
0.0533	0.3248
0.1260	0.5997
0.1350	-0.3960
0.2435	0.4054
0.3014	-0.6205
0.4464	0.7619
0.3143	1.1227
0.3739	-0.7886
0.1703	-0.9619
-0.0641	0.4440
-0.1198	0.6644
-0.0307	-0.9708
0.1424	0.5800
0.3601	-0.7714
0.6417	-0.9540
0.6444	1.3443

$\Sigma(d - d^{(\text{complet})})^2$	1,22E+11
--------------------------------------	----------

Tableau 7.3 : Estimation du  $d^{(\text{complet})}$ .

Itération 1 : Modèle 1 = Modèle complet – PIB industriel

Descripteurs	$\beta$
essence	1.0742
population	-1.6673
population active	1.7192
population urbaine	0.8011
dépenses ménages	-3.1645
parc auto	2.9418
dépenses gouvernement	-4.9947

Tableau 7.4 : coefficient MCO du modèle 1.



$d^{(incomplet)} * 10^6$	$(d - d^{(incomplet)}) * 10^5$
-1.2668	0.8979
-0.9632	-0.2599
-0.7802	-0.8618
-0.5323	0.1670
-0.1576	-0.8729
-0.0364	0.1285
0.0513	0.3444
0.1264	0.5959
0.1378	-0.4242
0.2432	0.4089
0.2959	-0.5658
0.4449	0.7772
0.3144	1.1217
0.3758	-0.8070
0.1786	-1.0446
-0.0637	0.4398
-0.1315	0.7816
-0.0363	-0.9149
0.1551	0.4527
0.3725	-0.8953
0.6272	-0.8087
0.6449	1.3392

$\Sigma(d - d^{(incomplet)})^2$	1,23E+11
---------------------------------	----------

F calculée	8,86E-02
F tabulée	244,56

Tableau 7.5 : Test de Fisher pour le modèle1.

Itération 2 : Modèle 2 = Modèle 1 – Dépenses ménages.

Descripteurs	$\beta$
essence	1.1396
population	-1.6326
population active	1.7176
population urbaine	0.7941
parc auto	2.7042
dépenses gouvernement	-4.9821

Tableau 7.6 : coefficient MCO du modèle 2.

$d^{(incomplet)} * 10^6$	$(d - d^{(incomplet)}) * 10^5$
-1.2556	0.7860
-0.9612	-0.2803
-0.7881	-0.7832
-0.5449	0.2937
-0.1662	-0.7861
-0.0434	0.1978
0.0661	0.1967
0.1365	0.4949
0.1410	-0.4565
0.2493	0.3481
0.3070	-0.6761
0.4230	0.9959
0.3052	1.2134
0.3883	-0.9319
0.1778	-1.0370
-0.0715	0.5181
-0.1301	0.7674
-0.0369	-0.9082
0.1609	0.3948
0.3708	-0.8778
0.6238	-0.7751
0.6483	1.3052
$\Sigma(d - d^{(incomplet)})^2$	1,24E+11
F calculée	1,40E-01
F tabulée	19,42

Tableau 7.7 : Test de Fisher pour le modèle2.

Itération 3 : Modèle 3 = Modèle 2 – Essence.

Descripteurs	$\beta$
population	1.3025
population active	1.5346
population urbaine	0.6606
parc auto	2.8583
dépenses gouvernement	-6.5851

Tableau 7.8 : coefficient MCO du modèle 3.

$d^{(incomplet)} * 10^6$	$(d - d^{(incomplet)}) * 10^5$
-1.2509	0.7386
-0.9680	-0.2125
-0.7534	-1.1301
-0.5387	0.2312
-0.2357	-0.0912
-0.0687	0.4515
0.0214	0.6438
0.1836	0.0245
0.2260	-1.3061
0.2870	-0.0291
0.3407	-1.0137
0.3651	1.5754
0.3065	1.2010
0.2805	0.1454
0.1641	-0.9000
0.0268	-0.4650
-0.1704	1.1709
-0.0224	-1.0532
0.1504	0.4999
0.3785	-0.9548
0.6515	-1.0519
0.6262	1.5254

$\Sigma(d - d^{(incomplet)})^2$	1,73E+11
---------------------------------	----------

F calculée	1,83E+00
F tabulée	8,73

**Tableau 7.9 : Test de Fisher pour le modèle3.**

Itération 4 : Modèle 4 = Modèle 2 – Population active.

Descripteurs	$\beta$
population	-1.8523
population urbaine	1.8571
parc auto	3.4844
dépense gouvernement	-4.8447

**Tableau 7.10 : coefficient MCO du modèle 4.**

$d^{(incomplet)} * 10^6$	$(d - d^{(incomplet)}) * 10^5$
-1.2575	0.8053
-0.9234	-0.6584
-0.7475	-1.1885
-0.5346	0.1909
-0.2334	-0.1143
-0.1341	1.1049
-0.0435	1.2928
0.1653	0.2070
0.2118	-1.1647
0.3337	-0.4963
0.4354	-1.9602
0.4112	1.1144
0.3107	1.1590
0.2424	0.5270
0.0915	-0.1738
-0.0475	0.2782
-0.1233	0.6997
0.0285	-1.5628
0.2127	-0.1231
0.4202	-1.3720
0.5923	-0.4597
0.5893	1.8945
$\Sigma(d - d^{(incomplet)})^2$	2,25E+11
F calculée	2,76E+00
F tabulée	5,89

Tableau 7.11 : Test de Fisher pour le modèle 4.

Itération 5 : Modèle 4 = Modèle 2 – Dépenses gouvernement à la consommation générale.

Descripteurs	$\beta$
population	-1.3951
population urbaine	0.9843
parc auto	4.3648

Tableau 7.12 : coefficient MCO du modèle 5.

$d^{(incomplet)} * 10^6$	$(d - d^{(incomplet)}) * 10^5$
-1.3314	1.5438
-0.9681	-0.2109
-0.7566	-1.0979
-0.5106	-0.0493
-0.1298	-1.1505
-0.0070	-0.1657
0.0667	0.1906
0.2229	-0.3685
0.2034	-1.0802
0.2495	0.3461
0.2960	-0.5662
0.2701	2.5247
0.1856	2.4095
0.1026	1.9243
0.0448	0.2933
0.0331	-0.5284
0.1241	-1.7743
0.2039	-3.3166
0.3070	-1.0660
0.4424	-1.5944
0.4516	0.9470
0.4998	2.7897
$\Sigma(d - d^{(incomplet)})^2$	4,95E+11
F calculée	7,97E+00
F tabulée	4,66

Tableau 7.13 : Test de Fisher pour le modèle 5.

Dans cette itération nous remarquons que :

$$F_{\text{calculée}} > F_{\text{tabulée}}$$

Donc le critère d'arrêt de la procédure est atteint, le modèle gasoil final comporte les facteurs suivants :

- Parc auto.
- Population.
- Population urbaine.
- Dépenses gouvernements à la consommation finale.

## 7.2. Le modèle essence :

:

années	d
1980	-687481.72727273
1981	-574230.72727273
1982	-506947.72727273
1983	-371662.72727273
1984	-185366.72727273
1985	-115608.72727273
1986	-3294.72727273008
1987	6889.27272726991
1988	61778.2727272699
1989	193517.27272727
1990	268563.27272727
1991	366932.27272727
1992	317388.27272727
1993	432902.27272727
1994	308633.27272727
1995	164372.27272727
1996	192609.27272727
1997	102610.27272727
1998	77833.2727272699
1999	22607.2727272699
2000	-11804.72727273
2001	-60238.72727273

Tableau 7.14 : l'évolution observée de la consommation annuelle de l'essence (centrée).

Itération 0 : Le modèle complet.

Descripteurs	$\beta$
gasoil	0.2296
population	0.5271
population active	-0.4419
population urbaine	-0.2566
PIB industriel	-0.0678
dépenses ménages	-2.6500
parc auto	-0.3414
Dépenses gouvernement	0.1745

Tableau 7.15 : coefficient MCO du modèle complet.

$d^{(\text{complet})} * 10^5$	$(d - d^{(\text{complet})}) * 10^4$
-6.7533	-1.2154
-5.8688	1.2652
-4.9737	-0.9583
-3.5106	-2.0600
-2.3940	5.4038
-1.2074	0.5130
-0.3724	3.3943
0.3962	-3.2730
1.0185	-4.0067
2.1963	-2.6118
2.6186	0.6699
3.6936	-0.2424
3.5248	-3.5089
3.3942	9.3478
2.8269	2.5940
2.4097	-7.6595
1.7718	1.5434
0.8718	1.5434
0.8508	-0.7248
0.1661	0.5993
-0.2040	0.8600
-0.4551	-1.4733
$\Sigma(d - d^{(\text{complet})})^2$	2,57E+10

Tableau 7.16 : Estimation du  $d^{(\text{complet})}$ .

Itération 1 : Modèle 1 = Modèle complet – Population urbaine.

Descripteurs	$\beta$
gasoil	0.1989
population	0.3579
pop active	-0.6459
PIB industriel MDA	-0.0690
dépenses ménages	-3.0533
parc auto	-0.1019
dépenses gouvernement	0.1580

Tableau 7.17 : coefficient MCO du modèle 1.

$d^{(incomplet)} * 10^5$	$(d - d^{(incomplet)}) * 10^5$
-6.8105	-0.0643
-5.7850	0.0427
-4.9197	-0.1498
-3.4931	-0.2235
-2.3692	0.5156
-1.3267	0.1706
-0.5247	0.4918
0.3569	-0.2880
1.0359	-0.4181
2.2876	-0.3525
2.8223	-0.1367
3.7788	-0.1095
3.5163	-0.3425
3.3198	1.0093
2.7094	0.3769
2.2435	-0.5998
1.7284	0.1977
0.9335	0.0926
0.9164	-0.1381
0.2867	-0.0606
-0.2057	0.0876
-0.5009	-0.1015
$\Sigma(d - d^{(incomplet)})^2$	2,75E+10
$F_{calculée}$	9,32E-01
$F_{tabulée}$	244,56

Tableau 7.18 : Test de Fisher pour le modèle 1.

Itération 2 : Modèle 2 = Modèle 1 – Population active.

Descripteurs	$\beta$
gasoil	0.0251
population	0.0858
PIB industriel	-0.2197
dépenses ménages	-5.7399
parc auto	0.4961
dépenses gouvernement	-1.3463

Tableau 7.19 : coefficient MCO du modèle 2.



$d^{(incomplet)} * 10^5$	$(d - d^{(incomplet)}) * 10^4$
-7.0256	1.5080
-5.8310	0.8874
-4.6752	-3.9429
-3.4425	-2.7409
-2.2808	4.2710
-1.2523	0.9621
-0.6400	6.0706
0.3190	-2.5014
1.2825	-6.6473
2.1999	-2.6476
2.7454	-0.5978
3.5990	0.7037
3.3671	-1.9323
3.3665	9.6254
3.0685	0.1781
2.3471	-7.0338
1.1629	7.6324
0.8711	1.5502
0.8723	-0.9396
0.6082	-3.8211
-0.0492	-0.6885
-0.6128	0.1042
$\Sigma(d - d^{(incomplet)})^2$	3,63E+10
F calculée	2,68E+00
F tabulée	19,42

Tableau 7.20 : Test de Fisher pour le modèle 2.

Itération 3 : Modèle 3 = Modèle 2 – Population.

Descripteurs	$\beta$
gasoil	-0.0032
PIB industriel	-0.1712
dépenses ménages	-8.7616
parc auto	0.9879
dépenses gouvernement	-1.0296

Tableau 7.21: coefficient MCO du modèle 3.

$d^{(incomplet)} * 10^5$	$(d - d^{(incomplet)}) * 10^5$
-7.3279	0.4531
-5.9134	0.1711
-4.6139	-0.4556
-3.2948	-0.4219
-1.9849	0.1312
-1.0084	-0.1477
-0.6274	0.5944
0.3122	-0.2433
1.2700	-0.6522
2.0925	-0.1573
2.5895	0.0961
3.7209	-0.0516
3.3024	-0.1285
3.1039	1.2251
2.9466	0.1397
2.3612	-0.7175
1.3417	0.5844
1.0365	-0.0104
0.8123	-0.0339
0.6327	-0.4066
-0.0306	-0.0875
-0.7212	0.1188
$\Sigma(d - d^{(incomplet)})^2$	4,12E+10
F calculée	2,61E+00
F tabulée	8,73

Tableau 7.22 : Test de Fisher pour le modèle 3.

Itération 4 : Modèle 4 = Modèle 3 – PIB industriel.

Descripteurs	$\beta$
gasoil	-0.0579
dépenses ménages	-7.7623
parc auto	1.0487
dépenses gouvernement	-1.6380

Tableau 7.23 : coefficient MCO du modèle 4.

$d^{(incomplet)} * 10^5$	$(d - d^{(incomplet)}) * 10^5$
-7.3186	0.4438
-5.8895	0.1472
-4.5720	-0.4975
-3.3404	-0.3763
-2.0100	0.1563
-1.0589	-0.0972
-0.6453	0.6124
0.2991	-0.2302
1.3128	-0.6951
2.1000	-0.1648
2.7273	-0.0417
3.6663	0.0030
3.2644	-0.0905
3.0909	1.2381
2.8962	0.1902
2.4107	-0.7670
1.3768	0.5493
1.1213	-0.0952
0.5151	0.2632
0.4591	-0.2331
0.3570	-0.4751
-0.7624	0.1600
$\Sigma(d - d^{(incomplet)})^2$	4,43E+10
F calculée	2,36E+00
F tabulée	5,89

Tableau 7.24 : Test de Fisher pour le modèle 4.

Itération 5 : Modèle 5 = Modèle 4 – Dépenses ménages.

Descripteurs	$\beta$
gasoil	-0.1039
parc auto	1.0118
dépenses gouvernement	-1.4488

Tableau 7.25 : coefficient MCO du modèle 5.

$d^{(incomplet)} * 10^5$	$(d - d^{(incomplet)}) * 10^5$
-7.4107	0.5359
-5.9722	0.2299
-4.7096	-0.3598
-3.5846	-0.1320
-2.0001	0.1464
-0.9660	-0.1900
-0.1018	0.0689
0.8692	-0.8003
1.6720	-1.0542
2.2948	-0.3597
3.0048	-0.3191
2.8213	0.8480
2.7570	0.4169
2.8922	1.4368
2.6101	0.4763
2.2290	-0.5852
1.4680	0.4580
1.3113	-0.2852
0.7376	0.0407
0.5871	-0.3610
0.2541	-0.3721
-0.7633	0.1609
$\Sigma(d - d^{(incomplet)})^2$	6,65E+10
F calculée	4,12E+00
F tabulée	4,66

Tableau 7.26 : Test de Fisher pour le modèle 5.

Itération 6 : Modèle 6 = Modèle 5 – Dépenses gouvernement à la consommation générale.

Descripteurs	$\beta$
gasoil	0.2695
parc auto	0.2530

Tableau 7.27 : coefficient MCO du modèle 6.

$d^{(incomplet)} * 10^5$	$(d - d^{(incomplet)}) * 10^5$
-6.0293	-0.8455
-5.0996	-0.6428
-4.4039	-0.6656
-3.0703	-0.6463
-1.8562	0.0025
-0.9229	-0.2332
-0.3539	0.3210
0.2068	-0.1380
0.1665	0.4513
0.9012	1.0339
0.9941	1.6915
1.9119	1.7574
1.7700	1.4038
1.5392	2.7898
1.0496	2.0368
0.9201	0.7236
0.9464	0.9797
0.8373	0.1888
1.8126	-1.0342
2.1546	-1.9285
2.9192	-3.0372
3.6066	-4.2089
$\Sigma(d - d^{(incomplet)})^2$	5,65E+11
F calculée	4,55E+01
F tabulée	3,98

Tableau 7.28 : Test de Fisher pour le modèle 6.

Dans cette itération nous remarquons que :

$$F_{\text{calculée}} > F_{\text{tabulée}}$$

Donc le critère d'arrêt de la procédure est atteint, le modèle essence final comporte les facteurs suivants :

- Gasoil.
- Parc auto.
- Dépenses gouvernements à la consommation finale.

**ANNEXE 8 :***Prévision des descripteurs*

Année	Population		
	série observée	série lissée	Résidus
1980	-1,67341907	-1,59820732	-0,07521175
1981	-1,51926361	-1,55894678	0,03968317
1982	-1,35864191	-1,38582277	0,02718087
1983	-1,1933645	-1,21220861	0,01884411
1984	-1,02187949	-1,03792372	0,01604423
1985	-0,84651472	-0,85876956	0,01225484
1986	-0,67865081	-0,67754699	-0,00110383
1987	-0,51647721	-0,51021071	-0,0062665
1988	-0,34990655	-0,35103249	0,00112594
1989	-0,18799159	-0,18392363	-0,00406796
1990	-0,02943908	-0,02395316	-0,00548593
1991	0,13118262	0,13197708	-0,00079445
1992	0,29361488	0,29221904	0,00139585
1993	0,45475389	0,45531851	-0,00056462
1994	0,61046124	0,61618763	-0,00572638
1995	0,7563399	0,76915777	-0,01281788
1996	0,88721684	0,90890948	-0,02169264
1997	1,01111024	1,02941734	-0,01830711
1998	1,13060658	1,14455994	-0,01395336
1999	1,24906832	1,25738658	-0,00831826
2000	1,36571951	1,37187219	-0,00615268
2001	1,4854745	1,4855824	-0,0001079
2002		1,60528582	
2003		1,72509713	
2004		1,84490845	
2005		1,96471977	
2006		2,08453108	
2007		2,2043424	
2008		2,32415371	
2009		2,44396503	
2010		2,56377634	

**Tableau 8.1 : Prévision de l'évolution de la population par la méthode du lissage exponentiel.**

Année	Population urbaine		
	série observée	série lissée	Résidus
1980	-1,52709127	-1,45113654	-0,07595473
1981	-1,39661589	-1,40497202	0,00835612
1982	-1,27035795	-1,27653611	0,00617817
1983	-1,13869147	-1,1472701	0,00857863
1984	-1,00656279	-1,01180653	0,00524373
1985	-0,86950247	-0,8770207	0,00751823
1986	-0,72712279	-0,73664459	0,0095218
1987	-0,57813408	-0,59000409	0,01187001
1988	-0,42986262	-0,43569331	0,00583069
1989	-0,28095809	-0,2843041	0,00334601
1990	-0,13082675	-0,13367909	0,00285234
1991	0,030647	0,0178052	0,0128418
1992	0,19661871	0,18451529	0,01210342
1993	0,35785097	0,35613301	0,00171796
1994	0,52652334	0,51888631	0,00763704
1995	0,68470821	0,69067426	-0,00596605
1996	0,85216208	0,84705378	0,0051083
1997	1,01824235	1,01609355	0,0021488
1998	1,18438326	1,18337418	0,00100909
1999	1,34435603	1,35006129	-0,00570526
2000	1,49722681	1,50786702	-0,01064022
2001	1,66300743	1,65616523	0,0068422
2002		1,82388462	
2003		1,98524077	
2004		2,14659692	
2005		2,30795306	
2006		2,46930921	
2007		2,63066535	
2008		2,7920215	
2009		2,95337765	
2010		3,11473379	

**Tableau 8.2 :Prévision de l'évolution de la population urbaine par la méthode du lissage exponentiel.**

Année	Parc auto		
	série observée	série lissée	Résidus
1980	-2,1436799	-2,06570679	-0,07797312
1981	-1,8258113	-2,00036531	0,17455401
1982	-1,55223806	-1,6868897	0,13465164
1983	-1,26111523	-1,31836476	0,05724954
1984	-0,89751632	-0,96154847	0,06403214
1985	-0,64465297	-0,5635888	-0,08106417
1986	-0,4389245	-0,29702838	-0,14189612
1987	-0,22091928	-0,14825296	-0,07266632
1988	-0,06787376	-0,00147416	-0,06639959
1989	0,10179131	0,11036744	-0,00857613
1990	0,2618654	0,25082174	0,01104366
1991	0,3777808	0,40909151	-0,03131071
1992	0,46549271	0,52456186	-0,05906915
1993	0,55816049	0,58958538	-0,03142489
1994	0,63752087	0,65241235	-0,01489148
1995	0,73011463	0,71619329	0,01392134
1996	0,81782274	0,80480277	0,01301997
1997	0,88643433	0,90045316	-0,01401883
1998	0,95474033	0,97224907	-0,01750874
1999	1,04425351	1,03184476	0,01240875
2000	1,0855632	1,116032	-0,0304688
2001	1,13119101	1,15760501	-0,026414
2002		1,18618732	
2003		1,23011617	
2004		1,27404502	
2005		1,31797386	
2006		1,36190271	
2007		1,40583156	
2008		1,44976041	
2009		1,49368925	
2010		1,5376181	

**Tableau 8.3 : Prévision de l'évolution de la Parc auto par la méthode du lissage exponentiel.**



Année	Dépenses gouvernement		
	série observée	série lissée	Résidus
1980	-0,92695252	-0,85760416	-0,06934836
1981	-0,90676073	-0,7882558	-0,11850493
1982	-0,88416501	-0,76806401	-0,116101
1983	-0,86349229	-0,74546828	-0,11802401
1984	-0,83945437	-0,72479557	-0,1146588
1985	-0,81157039	-0,70075765	-0,11081274
1986	-0,77214829	-0,67287367	-0,09927462
1987	-0,74330276	-0,63345157	-0,10985119
1988	-0,70772666	-0,60460603	-0,10312062
1989	-0,67936184	-0,56902994	-0,1103319
1990	-0,61638235	-0,54066511	-0,07571723
1991	-0,44811712	-0,47768562	0,0295685
1992	-0,23850607	-0,3094204	0,07091433
1993	-0,07415143	-0,09980935	0,02565792
1994	0,23600301	0,06454529	0,17145772
1995	0,5585919	0,37469974	0,18389216
1996	0,97781274	0,69728862	0,28052412
1997	1,17684618	1,11650946	0,06033672
1998	1,37491916	1,3155429	0,05937625
1999	1,55472284	1,51361588	0,04110696
2000	1,64751734	1,69341956	-0,04590222
2001	1,98567866	1,78621407	0,19946459
2002		2,12437538	
2003		2,2630721	
2004		2,40176882	
2005		2,54046555	
2006		2,67916227	
2007		2,81785899	
2008		2,95655571	
2009		3,09525244	
2010		3,23394916	

**Tableau 8.4 : Prévision de l'évolution de la Dépenses gouvernement par la méthode du lissage exponentiel.**

# Références bibliographiques

# *Références bibliographiques :*

- [ARB 03] : M.ARBIB.  
The hand book of brain and neural networks.  
The MIT press, 2003.
- [BER 01] : K. BERTELS, L. NEUBERG & S. VASSILIADIS.  
On Chaos and Neural Networks: The Backpropagation Paradigm.  
Artificial Intelligence Review, 2001.
- [BER 94] : H.BERSINI, M.SAERENS & L.G.SOTELINO.  
Hopfield net generation, encoding and classification of temporal trajectories.  
IEEE transactions on neural networks, vol 13, 1994.
- [BOU 01] : R.BOURBANNAIS & J.C.SINIER.  
Prévision des ventes : Théories et pratiques.  
Ed. ECONOMICA collection Gestion, 2001.
- [COU 99] : P.COULIBALY, F.ANCTIL & B.BOBEE.  
Prévision hydrologique par réseaux de neurones artificiels : état de l'art.  
CNRC CANADA, Vol 26, pp 293-304, 1999.
- [COU 97] : F.COULON.  
Mise en oeuvre d'un réseau de neurones Récurrents dynamiques sur une Architecture  
parallèle dédiée A mémoire distribuée.  
PFE: faculté polytechnique de mons, 1997.
- [DAV 92] : E.DAVALO & P.NAIM.  
Des réseaux de neurones.  
Ed. Eyrolles, 1992.
- [DAZ 01] : B.A.DAZSI & R.ENBODY.  
Artificial Neural Networks For Branch Prediction.  
Master of science, Michigan State University, 2001.
- [DEC 03] : C.DECAESTECKER & M.SAERENS.  
Réseaux de neurones artificiels.  
LINF, 2003.
- [DIL 91] : T.S. DILLON & S. LEUNG.  
An adaptative neural network approach in load forecasting in a power system.  
IEEE Transactions on Power Systems, Vol 11, pp 7-11, 1991.
- [DRE 04] : G.DREYFUS.  
Réseaux de neurones : méthodologie et application.  
Ed. Eyrolles, 2004.

- [FAU 98] : A.FAURE.  
Cybernétique des réseaux neuronaux : commande et perception.  
Ed. HERMES, 1998.
- [FIO 99] : A.FIORDALISO.  
Système flous et prévision de séries temporelles.  
Ed. HERMES Science Publications, 1999.
- [GAW 03] : A.E.GAWEDA, A.JACOBS, M.E.BRIER & J.M. ZURADA.  
Pharmacodynamic population analysis in chronic renal failure using artificial neural networks- a comparative study.  
ELSEVIER, NEURAL NETWORKS, Vol 16, pp 841-845, 2003.
- [GIA 03] : V.GIARD.  
Statistique appliquée à la gestion 8<sup>ème</sup> édition.  
Ed. Gestion & Economica, 2003.
- [GUY 90] : A.GUY.  
Les méthodes de prévision en économie.  
Ed. ARMAND COLIN, 1990.
- [HAM 90] : C.HAMMOND & D.TRITSCH.  
Neurobiologie cellulaire.  
Ed. DOIN, 1990.
- [HOW 00] : D.HOWARD & M.BEALE.  
Neural Network Toolbox For use with MATLAB Version 4.  
Ed. THE MATHWORKS, 2000.
- [IWA 99] : H.IWAMIYA & B.KERMANSHAHI.  
Long-term Load Forecasting using Neural Nets.  
IEEE Transactions on Power Systems, Vol 12, pp 13-17, 1999.
- [JAN 02] : N.JANKOWSKI & V.KADIRKAMANATHAN.  
Statistical control of growing and pruning in RBF-like neural networks.  
IEEE transactions on neural networks, vol 3, pp 412-416, 2002.
- [KAA 95] : I.KAASTRA & M.BOYD.  
Designing a neural network for forecasting financial and economic time series.  
ELSEVIER & NEUROCOMPUTING, 1995.
- [KHE 97] : V.H.KHER & S.K.JOSHI.  
Short term electrical load forecasting using artificial neural network.  
IEEE Transactions on Power Systems, Vol 27, pp18-23, 1997.
- [LAD 03] : R.LADJADJ.  
Les réseaux de neurones.  
Ed. Ingénieur 2000, 2003.

- [LEE 91] : K.Y.LEE & Y.T.CHA & C.C.KU.  
A study on neural networks for short term load forecasting.  
IEEE Transactions on Power Systems, Vol 14, pp 26-30, 1991.
- [MEL 90] : G.MELARD.  
Méthodes de prévision à court terme.  
Ed. ELLIPSES, 1990.
- [PAR 04] : M.PARIZEAU.  
Réseaux de neurones.  
Université LAVAL, 2004.
- [RIC 01] : D.RICHARD & D.ORSAL.  
Neurophysiologie : organisation et fonctionnement du système nerveux.  
Ed. DUNOD, 2001.
- [ROL 96] : F.ROLI, S.B.SERPICO, L.BRUZZONE & G.VERNAZZA.  
Classification of multispectral remote-sensing images by neural networks.  
IEEE trans on geosc and remote sens, Vol 31, pp 364-367, 1996.
- [SEN 96] : A.SENOUSSE & M.T.BOURBON.  
Elaboration d'un modèle de prévision de la charge électrique nationale en temps réel.  
Mémoire de fin d'étude, ENP, Génie industriel. 1996.
- [STU 05] : W.STUETZLE.  
Gram-Schmidt orthogonalization and stepwise regression.  
BioStat, 2005.
- [SRI 91] : D.SRINIVASAN & A.C.LIEW & J.S.P.CHEN.  
Short term forecasting using neural network approach.  
IEEE Transactions on Power Systems, Vol 13, pp12-16, 1991.
- [TOU 92] : C.TOUZET.  
Les réseaux de neurones Artificiels : Introduction au Connexionnisme.  
E-books, 1992.
- [VER 99] : M.VERLEYSSEN, E.de BODT & A.LENDASSE.  
Forecasting financial time series through intrinsic dimension estimation and non-linear data projection.  
International Work-conference on Artificial and Natural Neural Networks,  
Lecture Notes in Computer Science 1607, pp 596-605, 1999.
- [XIN 02] : Y.XINGHUO, M.ONDER EFE & K.OKYAY.  
A General Backpropagation Algorithm for Feedforward Neural Networks.  
IEEE transactions on neural networks, vol 13, 2002.

## Ouvrages et articles consultés :

- [CRO 02] : S.F.CRONE.  
Stepwise Selection of Artificial Neural Networks Models for Time Series Prediction.  
Lancaster University Management School, 2002.
- [DOR 03] : S.DORLING, G.CAWLEY, J.KUKKONEN & A.KARPPINEN.  
Air Quality Forecasting Using Neural Network Approaches.  
UEA ENV, 2003.
- [FRE 91] : J.A.FREEMAN & D.M. SKAPURA.  
Neural Networks : Algorithms, Applications, and Programming Techniques.  
Addison-Wesley Publishing Company, 1991.
- [GAV 01] : M.GAVRILAS.  
Neural network based forecasting for Electricity markets.  
Technical University of Iasi, 2001.
- [HAY 01] : S.HAYKIN.  
Kalman Filtering and Neural Networks.  
John Wiley & Sons, 2001.
- [JAI 96] : A.K.JAIN, J.MAO & K.M.MOHIUDDING.  
Artificiel neural networks : a tutorial.  
IEEE, Vol 29, pp 31-44, 1996.
- [KAS 98] : N.K.KASABOV.  
Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering,      **Second**  
edition.  
The MIT Press, 1998.
- [LEN 00] : A.LENDASSE, E.de BODT, V.WERTZ & M.VERLEYSEN.  
Non-linear financial time series forecasting –Application to the Bel 20 stock market index.  
European Journal of Economic and Social Systems, Vol 14, pp 81-91, 2000.
- [MAR 02] : W.L.MARTINEZ & A.R.MARTINEZ.  
Computational statistics handbook with MATLAB.  
CHAPMAN & HALL / CRC, 2002.
- [NEE 94] : P. S.NEELAKANTA & D.DEGROFF.  
Neural Network Modeling: Statistical Mechanics and Cybernetic Perspectives.  
CRC Press, 1994.
- [PER 98] : M.PEREZ.  
Neural networks applications in bankruptcy forecasting :a state of the art.  
Centre de Recherche de l'IAE, Université Jean Moulin Lyon 3, 1998.

- [SFI 04] : C.V.SFINTE, M.GAVRILA & O.Ivanov.  
Application of heat index for electricity load Forecasting.  
WEC Regional Energy Forum, 2004.
- [SIN 00] : S.SINGH.  
Pattern modelling in time-series forecasting.  
Cybernetics and Systems-An International Journal, Vol 31, pp 1-25, 2000.
- [SOA 04] : J.SOARES, A.P.OLIVEIRA, M.ZLATA, B.ZNAR, P.MLAKAR & J.F.ESCOBEDO.  
Modeling hourly diffuse solar-radiation in the city of Sao Paulo using a neural-network  
technique.  
ELSEVIER & APPLIED ENERGY, Vol 79, pp 2001-2014, 2004.
- [SUB 98] : C.SUBRAMANIAN, M.T. MANRY & J.NACCARINO.  
Reservoir inflow forecasting using neural networks  
IEEE, Vol 13, pp 12-17, 1998.
- [VAN 96] : P.VAN DER SMAGT & B.KRÖSE.  
An introduction to neural networks. Eighth edition.  
The university of AMSTERDAM, 1996.
- [WIL 02] : I.WILSON, S.PARIS, A.WARE & D.JENKINS.  
Residential Price Forecasting at National and Regional Levels.  
XXII International Congress Washington, D.C, 2002.

## Sites web:

[www.grappa.univ-lille3.fr](http://www.grappa.univ-lille3.fr)

Apprentissage automatique : les réseaux de neurones.

[www.isib.be](http://www.isib.be)

Le perceptron.

[www.crteknologies.free.fr](http://www.crteknologies.free.fr)

Prévision de température par réseaux de neurones.

[www.membres.lycos.fr/villemingerard/Logique/IAneuron.html](http://www.membres.lycos.fr/villemingerard/Logique/IAneuron.html)

Intelligence artificielle réseaux de neurones neuronique.

[www.neural-forecasting.com](http://www.neural-forecasting.com)

Faire des prévisions avec les réseaux de neurones

[www.afdb.org](http://www.afdb.org)

Source de données.

## ملخص :

تعتبر التنبؤات إحدى أهم وسائل اتخاذ القرار و هي الآن أكثر تعقيدا نظرا للتطورات اللاحقة للظواهر الاقتصادية.

من خلال هذه الدراسة استعملنا طريقة جديدة للتنبؤ و المتمثلة في الشبكات العصبونية الاصطناعية و على وجه الخصوص البرسيبترون المتعدد الطبقات (ب م ط)، الذي استخدمناه في تنبؤ استهلاك الوقود (المازوت و البترين).

الكلمات المفتاحية : التنبؤات، الشبكة العصبونية، ب م ط ، الوقود.

## **Résumé :**

La prévision, outil important d'aide à la décision, devient de plus en plus difficile vu les non-linéarités des phénomènes économiques.

A travers le présent travail, nous nous sommes intéressés à une nouvelle méthode de prévision qui est les réseaux de neurones artificiels et en particulier les perceptrons multicouches (PMC) que nous avons appliqué à la consommation des carburants (gasoil et essence) à l'horizon 2010.

*Mots clés :* prévision, réseaux de neurones, PMC, carburant.

## **Abstract :**

The forecast, tool significant of decision-making aid, becomes increasingly difficult considering non-linearities of the economic phenomena.

Through this work, we were interested in a new method of forecast which is the artificial neural networks and in particular the multi-layer perceptrons (MLP) that we applied to the consumption of the fuels (gas oil and gasoline) by 2010.

*Key words:* forecast, neural networks, MLP, fuel.