

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieurs et la recherche scientifique

Université IBN KHALDOUN – Tiaret



Département de l'informatique

MEMOIRE POUR L'OBTENTION DU DIPLOME DE MASTER

EN INFORMATIQUE

OPTION : GENIE LOGICIEL

INTITULE :

Application des techniques intelligentes pour la prédiction des propriétés de PEBD

GHAZI Abdelkarim

CHAIB Ala Eddine

Encadreur :

Mr. AID Lahcene

Année universitaire : 2015-2016

Remerciement

*Nous remercions notre dieu **ALLAH** qui nous a donné la santé, le courage, la force et la patience d'achever ce modeste travail.*

*Avec beaucoup de respect et d'estime que nous remercions notre encadreur monsieur **AID Lahcene** d'avoir proposé, dirigé et suivi ce travail.*

Nous tenons aussi à remercier nos collègues pour leurs précieux conseils et leur disponibilité durant toute la période de notre projet.

Nous tenons à adresser nos remerciements à tous les enseignants de département d'informatique.



Je dédie cet humble travail à toute ma famille je cite particulièrement :

- A ma mère et mon père
- A ma sœur & mon frère
- A tous mes ami(e)s et surtout Annan Anas, Chouaf Mohamed
- A tous les profs que j'ai rencontrés dans mon parcours
- A tout ce monde je dis Merci

Ghazi Abdelkrim



Je dédie cet humble travail à toute ma famille je cite particulièrement :

- A ma mère et mon père
- A ma sœur & mon frère
- A tous mes ami(e)s et surtout Annan Anas, Chouaf Mohamed ,BOUKHORS Fethi , Arbia younes
- A tous les profs que j'ai rencontrés dans mon parcours
- A tout ce monde je dis Merci

Chaïb Ala Eddine

Résumé :

Les films de polyéthylène sont utilisés dans la plasticulture pour favoriser la culture protégée. Grâce à la Co-extrusion, qui s'est présentée comme une nouvelle technique dans ces dernières années, on peut associer plusieurs couches de polymères entre elles. Chacune de ces couches possède ses qualités propres ; résistance accrue, amélioration de l'aspect, tenue au froid, amélioration de la soudure ou du scellage, propriétés barrières. Les films sont opposés aux conditions climatiques qui participent à leurs dégradations et vieillissements.

Le projet porte sur la compréhension et l'étude de ces phénomènes de dégradation dans les climats sévères (cas de la région de Tiaret) par simulation numérique. Cette étude sert à optimiser ces films pour allonger leur durée de vie, réduire leur coût, et ainsi estimer les paramètres climatiques influents sur leur dégradation.

Mots clés : PEBD, Réseaux de neurones, modélisation, simulation.

Abstract:

The polyethylene is used in plasticulture for protected cultivation. Thanks to the co- extrusion, which is presented as a new technique in recent years, one can combine several layers of polymers together. Each layer has its own qualities; increased strength, improved appearance, resistance to cold, improving the welding or sealing, barrier properties. The films are opposed to the climatic conditions involved in their degradation and aging.

The project involves the understanding and study of these phenomena of degradation in severe climates (as in Tiaret region) by numerical simulation. This study serves to optimize these films to lengthen their lifespan, reduce cost, and thus estimate the influential climatic parameters on degradation.

Keywords: LDPE, neural networks, modeling, simulation.

Sommaire

Contents

Introduction générale :	1
I.1. Introduction :	3
I.2. Définition :	3
I.3. Classification des polymères :	3
I.3.1. Selon leur Origine :	3
I.3.1.1. Les polymères naturels	4
I.3.1.2 Les polymères artificiels.....	4
I.3.1.3. Les polymères synthétiques.....	4
I.3.2 Selon leur Domain d'application.....	4
I.3.2.1. Les polymères de Grande diffusion :	4
I.3.2.2. Les polymères techniques.....	4
I.3.2.3. Les polymères spéciaux (ou polymères de fonction)	5
I.4. Propriétés des polymères :	5
I.4.1. Propriétés physiques des polymères :	7
I.4.2. Propriétés thermiques :	7

I.4.2.1. La température :	7
I.4.2.2. Comportement mécanique et thermique :	8
I.4.3. Propriétés mécaniques :	9
I.5. Généralités sur le vieillissement des polymères semi cristallins :	11
I.6. Déformation de la phase amorphe :	11
I.7. Les facteurs affectant les polymères :	12
I.7.1. La teneur relative en eau :	12
I.7.2. La température :	12
I.7.3. Les rayonnements ultra-violet (UV) :	13
I.7.4. Le pH :	14
I.8. Présentation du Polyéthylène à basse densité (PEBD) :	15
I.9. Structure du polyéthylène :	15
I.10. Structure moléculaire (échelle D):	17
I.11. Conclusion :	17
II.1. Introduction.....	19
II.2 Pourquoi les réseaux de neurones ?.....	19
II.3 Généralités	20
II.3.1 Neurone biologique	20
II.3.2 Neurone formel.....	21
II.3.3 Fonction d'activation.....	22

II.3.4	Définition d'un réseau de neurones.....	22
II.4	Réseaux de neurones pour la régression non-linéaire.....	23
II.4.1	La régression non linéaire.....	23
II.4.2	Modèle non linéaire par rapport aux variables.....	24
II.5	Architecture des réseaux de neurones.....	25
II.5.1	Les réseaux de neurones non bouclés (réseau statique).....	25
II.5.2	Les réseaux de neurones bouclés (réseau dynamique).....	26
III.5.3	Taxonomie des réseaux de neurones (non exhaustive).....	28
II.6	Le perceptron multicouche MLP.....	28
II.6.1	Propriétés d'approximations du MLP.....	30
II.7	Les réseaux RBF.....	30
II.8	L'apprentissage des réseaux de neurones.....	31
III.8.1	L'apprentissage supervisé.....	32
II.8.1.1	L'ensemble d'exemple d'apprentissage.....	33
II.8.1.2	Normalisation des entrées et des sorties.....	33
II.8.1.3	Fonction de coût.....	34
II.8.1.4	Algorithme d'optimisation.....	35
II.8.1.5	Calcul du gradient par rétropropagation de l'erreur.....	36
II.8.1.6	Modification des paramètres.....	38
II.8.1.7	Résumé de la rétropropagation du gradient.....	40

II.8.2	L'apprentissage non supervisé.....	42
II.9	Apprentissage et généralisation	42
II.9.1	Généralisation par un arrêt prématuré.....	43
II.9.2	Généralisation par régularisation.....	43
II.9.3	Régularisation par modération des poids (Weight-decay)	44
II.10	Sélection de modèle	45
II.10.1	La validation croisée (cross-validation)	45
II.10.2	La méthode Leave-one-out.....	46
II.11	Conclusion.....	46
III.1.	Introduction :.....	48
III.2.	Les choix techniques :.....	48
III.2.1.	Matlab :	48
III.2.2.	C# :	49
III.2.3.	SharpDevlope:.....	49
III.3.	Les interfaces de l'application :	50
III.3.1.	Création d'un réseau du neurone :	50
III.3.2.	Sauvegarde un réseau du neurone :.....	52
III.3.3.	Modifier les paramètres de réseau de neurone :.....	52
III.3.3.1.	Modifier les poids :	52
III.3.3.2.	Modifier les valeurs min/max des entrées et de la sortie :.....	53

III.3.4. L'apprentissage :	54
III.4. Les résultats :	57
III.4.1. Les tests :	57
III.4.1. Prediction:	59
III.5. Conclusion :	61
Conclusion générale :	62
Bibliographie :	64
Les paramètres de réseau	75

Table des figures

Figure I.1 : Evolution du module d'élasticité d'un polymère amorphe en fonction de la température : cas de macromolécule monodimensionnelle	8
Figure I.2 : Différentes sollicitations mécaniques.	9
Figure I.3 : Comportement d'un matériau soumis à une contrainte mécanique.	10
Figure I.4: Mécanisme de déformation dans la phase amorphe interlamellaire	12
Figure I.5 : Le spectre d'émission solaire. (La surface noire est filtrée par la couche d'ozone et la ligne pointillée représente l'intensité de lumière distribuée à la surface terrestre).....	14
Figure I.6 : Représentation semi-développée de la chaîne macromoléculaire du polyéthylène. ..	16
Figure I.7 : Représentation de la polymérisation du polyéthylène.	17
Figure II.1 : Neurone biologique.	21
Figure II.2 : Neurone formel.....	22
Figure II.3 : Les principales fonctions d'activation : (a) fonction logistique, (b) fonction tangente hyperbolique, (c) fonction linéaire, (d) fonction à seuil	23
Figure II.4 : Courbe de régression non linéaire pour des données.....	24
Figure II.5 Réseaux de neurones feedforward [PAL 06]......	27
Figure II.6 : Exemple d'un réseau de neurones bouclé. Les nombres dans les carrés sont les retards (exprimés en nombre de périodes d'échantillonnage) associés à chaque connexion [DRE 02].	27

Figure II.7 : La forme canonique du réseau de la Figure II.6 [DRE 02].	28
Figure II.8 : réseaux MLP à une couche cachée	29
Figure II.9 : Réseau RBF.	31
Figure II.10 : Exemple de minima locaux	34
Figure II.11 : l'évolution de la fonction de coût [DU 06].	43
Figure II.12 : Principe de la validation croisée	46
Figure III.1 : logo MATLAB	48
Figure III.2 : logo C#	49
Figure III.3: logo SharpDevelop	49
Figure III.4 : création d'un nouveau réseau de neurone	50
Figure III.6 : Ouvrir un nouveau réseau de neurone	51
Figure III.7: Sauvegarde un nouveau réseau de neurone	52
Figure III.8: fenêtre de modification des poids et biais	53
Figure III.9 : fenêtre de modification des min/max	53
Figure III.10 : fenêtre d'apprentissage	54
Figure III.11 : fenêtre de chargement d'un fichier Excel.	55
Figure III.12 : fenêtre de prédiction	56
Figure III.13 : fenêtre de prédiction	56
Figure III.14 : graphe de test pour temps =120.	57
Figure III.15 : graphe de test pour temps =1010.	58

Figure III.16 : graphe de test pour strain =1010	58
Figure III.17 : graphe de test pour strain =200	59
Figure III.18 : graphe de prédiction pour temps =300	59
Figure III.19 : graphe de prédiction pour temps =60	60
Figure III.20 : graphe de prédiction pour strain =500.....	60
Figure III.21 : graphe de prédiction pour strain =700.....	61

Introduction générale :

Les polymères sont des matériaux organiques qui se caractérisent par une grande facilité de mise en œuvre et d'excellents rapport propriétés-poids et propriétés-prix [1]. Ce sont ces raisons qui font leur succès dans la plasticulture, le concurrent principal étant le verre de silice.

La plasticulture n'est pas à proprement parler une science, mais un ensemble de techniques basée sur toutes les sciences intéressent la production agricole : l'agronomie, la climatologie, l'hydraulique, la physiologie végétale et particulièrement la science des polymères. Le polyéthylène fait partie des matériaux les plus utilisés en plasticulture.

Les principales propriétés qui ont assuré le succès du polyéthylène sont sa légèreté et sa transparence. Cependant, et malgré sa bonne inertie chimique, le polyéthylène, comme toutes les substances organiques, se dégrade peu à peu sous l'influence conjuguée, de la chaleur, du rayonnement solaire ultraviolet, des sollicitations mécaniques et des agents chimiques.

Si les mécanisme complexes de cette dégradation restant mal connus [2]. On dispose actuellement d'un nombre important de produit qui, additionnés à faible concentration aux polymères, les rendent plus stables à la chaleur et rayonnement ultraviolet [3]. C'est ainsi qu'il existe aujourd'hui un nombre considérable de formation possible de films en polyéthylène, afin d'adapter leurs propriétés à leurs usages. Dilara et briassoulis ont tenté de rassembler et de comparer les normes concernant l'utilisation des films plastique en couverture des serres.

La durée d'utilisation des matériaux plastique est relativement courte par rapport à leur durée de vie. De quelques années suivant les conditions d'usage et environnement.

La dégradation des polymères en utilisation extérieure est essentiellement due aux radiations UV du spectre solaire [4]. La photo dégradation constitue l'aspect le plus important du vieillissement climatique. Elle est généralement accentuée, par l'agressivité naturelle des conditions climatiques [5] liées Au site géographique d'exposition des diminutions des propriétés mécaniques et une altération des propriétés optique.

Des travaux expérimentaux font l'objet de l'étude du comportement des films multicouches où s'accumule température, UV et la pluie qui sont des éléments responsables du processus de vieillissement des serres agricoles dans la région de Tiaret. Dans ces travaux, les films multicouches ont subi différents vieillissements, artificiel durant sept mois (le tri couches Agro film) et vieillissement artificiel par température à 40°C avec l'eau, 50°C avec l'eau, 25°C avec et sans l'eau et 10°C avec et sans l'eau sous l'UV-A.

Vu le coût de ces expériences en matière de temps et de moyens, et la multitude des facteurs à tenir en compte, nous voulons dans ce travail utiliser la simulation informatique comme un support pour l'étude du vieillissement de ces films. Les réseaux de neurones se présentent comme le formalisme adéquat pour tenir tous ces paramètres en compte. Donc, ce travail consiste à l'étude du vieillissement des films plastique multicouches à l'aide de la simulation informatique basée sur les réseaux de neurones.

Chapitre 1 :

Les polymères

I.1. Introduction :

Les polymères sont omniprésents en odontologie, que ce soit en odontologie conservatrice comme matériaux de reconstitution ou en prothèses comme matériaux d'infrastructure, cosmétiques ou à empreinte. Ils rentrent également dans la constitution de cires ou des fils de sutures. Les mécanismes d'action de ces polymères au contact des différents tissus bucco-dentaires font appel à une stratégie chimique de plus en plus élaborée pour tenter de répondre convenablement et durablement à des conditions cliniques variées.

Pour mettre en œuvre correctement ces matériaux, il est important de connaître leur mécanisme de réaction avec leurs avantages et leurs inconvénients spécifiques.

Le but de ce chapitre est d'expliquer clairement et simplement la chimie macromoléculaire intéressant notre discipline et de permettre ainsi aux étudiants et aux praticiens d'apprécier les progrès technologiques et l'évolution des composites et autres matériaux de reconstitution cosmétiques ainsi que des matériaux prothétiques.

I.2. Définition :

La définition d'un polymère est : Produit constitué de molécules caractérisées par un grand nombre de répétitions d'une ou de plusieurs espèces d'atomes ou de groupes d'atomes (motifs constitutionnels) reliés entre eux en quantité suffisante pour conduire à un ensemble de propriétés qui ne varient pratiquement pas avec l'addition ou l'élimination d'un seul ou d'un petit nombre de motifs constitutionnels. [6]

I.3. Classification des polymères :

Il existe plusieurs modes de classification des polymères que peuvent être classés selon divers critères :

I.3.1. Selon leur Origine :

On distingue : [7]

I .3.1.1. Les polymères naturels

Sont issus du règne végétal où l'animal, leur importance est considérable. On peut cependant mentionner, dans cette catégorie, la famille des polysaccharides (cellulose, amidon...), le caoutchouc naturel, etc.

I .3.1.2 Les polymères artificiels

Sont obtenus par modification chimique de polymères naturels, de façon à transformer certaines de leurs propriétés ; l'ester cellulosique (nitrocellulose, acétate de cellulose...), ont toujours connu une certaine importance économique.

I .3.1.3. Les polymères synthétiques

Totalement issus du génie de l'homme, sont obtenus par polymérisation de molécules monomères. Leur variété est extrême et ce sont eux qui seront le plus souvent considérés par la suite.

I .3.2 Selon leur Domain d'application

Il est difficile de proposer une classification exhaustive tant la variété des propriétés a multiplié les applications des polymères, comme matériaux en particulier. Il est cependant possible de regrouper les polymères en trois grandes catégories [08]

I .3.2.1. Les polymères de Grande diffusion :

Appelés (polymères de commodité) Dont la production annuelle s'évalue par millions de tonnes, sont devenus d'un emploi quotidien pour tous. Le polyéthylène le polystyrène, le poly (chlorure de vinyle) et quelques autres sont à classer dans cette catégorie d'une importance économique considérable.

I .3.2.2. Les polymères techniques

Ont des caractéristiques qui leur permettent de se substituer, de plus en plus aux matériaux traditionnels (métaux, céramiques...) pour de nombreuses applications ; les polyamides, les poly acétals... font partie de cette famille.

I.3.2.3. Les polymères spéciaux (ou polymères de fonction)

Présentent généralement une propriété qui induit leur utilisation pour une application particulière. C'est dans cette catégorie se trouvent les polymères conducteurs, photo actifs, thermostables, adhésifs, etc.

Tous les spécialistes ne donnent pas la même définition à chacune de ces catégories même s'ils s'accordent sur les termes.

I.4. Propriétés des polymères :

Les propriétés des matériaux polymères sont beaucoup plus sensibles que celles des métaux aux influences extérieures telles que la température, la dureté, l'intensité de la contrainte appliquée, les radiations UV et les agents chimiques.

Les caractéristiques de résistance peuvent différer seulement d'un ordre de grandeur. Hormis les paramètres spécifiques ou matériaux (masse molaire, taux de ramification, mobilité des chaînes, taux de réticulation, ...), et les conditions extérieures (humidité, agents chimiques, température, vitesse de sollicitation, type et intensité des contraintes appliquées). [09]

	Propriétés	Normes	PEhdr	PEbdl	Unités
Physique	Masse volumique	NF T 51-063	915 à 940	917 à 940	Kg/m ³
	Indice de fluidité	NF T 51-016	0.2 à 70	3 à 8	g/10min
	Absorption d'eau	NF T 51-002	<0.03	<0.03	%
	Perméabilité x10 ⁵				m ³ /(Pa.s)
	H ₂ O		4.3	Moins perméable que le PEbdr	
	O ₂		0.2		

	CO ₂		0.7		
	N ₂		0.06		
Mécanique	Contrainte rupture	NF T 51-034	9à23	13 à 24	Mpa
	Allongement rupture	NF T 51-034	150 à 1000	600à 1200	%
	Module traction	NF T 51-034	200 à 500	220 à 600	Mpa
	Module flexion	NF T 51-034	60 à 400		Mpa
	Résistance aux chocs		350 à 1000	48 à 63	Mj/mm ²
	Dureté shore	NF T 51-019	40 à 47		
Thermique	Température de fusion	NF T 51-621	10 à 120	90	°C
	Température maximale		80 à 85		°C
	Capacité thermique		3200		USI
	Conductibilité thermique	ASTMC177-76	0.35		USI
	Coefficient de dilatation		2.3 10-4		USI
	Retrait moulage	ASTMD696-70	1.5 à 3.5		%
Electrique	Permittivité à 1 MHZ	NF C 26-230	2.28 à 2.3	2.28 à 2.3	
	Rigidité diélectrique	NF C 26-225	≡ 65	≡ 65	KV/mm ²
	Résistivité transversale	NF C 26-225	1018	1018	W.cm

TAB I.1 : Ordres de grandeurs des propriétés du PE.

I .4.1. Propriétés physiques des polymères :

Avant tout, rappelons qu'il existe une grande variété de matières plastiques, tout comme il existe un grand nombre d'alliages métalliques, une des caractéristiques physiques générales des polymères est :

La masse volumique des matières plastiques est peu élevée. La légèreté des polymères est sans aucun doute une des qualités qui a le plus largement contribué à leur diffusion.

En ce qui concerne le rapport (résistance à la traction / masse volumique), certains polymères sont en fait supérieurs bien à des matériaux métalliques.

La faible masse volumique des plastiques est due au faible poids atomique des principaux atomes de leurs chaînes (principalement l'hydrogène et le carbone). [09]

I .4.2. Propriétés thermiques :

I .4.2.1. La température :

La température de transition vitreuse "Tg" et la température de fusion "Tf" sont les deux températures fondamentales nécessaires dans l'étude des matériaux polymères. La température de transition vitreuse est partiellement importante pour les polymères amorphes, notamment les thermoplastiques amorphes, pour lesquels, il n'existe aucune force de cohésion importante autre que l'enchevêtrement. [11].

Les températures caractéristiques d'un seul et même matériau peuvent alors être classées de la façon suivante :

- Température de transition vitreuse
- Température de cristallisation
- Température de fusion
- Température de décomposition thermique. [11]

Selon la température à laquelle il est soumis, un matériau polymère peut présenter des comportements mécaniques différents. Ceci peut se produire pour les thermoplastiques semi cristallins dans un domaine même étroit de la température. [11]

I.4.2.2. Comportement mécanique et thermique :

Les propriétés mécaniques des polymères dépendent fortement de la température (Figure I.2). A basse température, la plupart des polymères présentent un état vitreux : ils sont rigides. Lorsque la température augmente, ils passent par un état de transition : pour une plage de températures spécifique du polymère, les chaînes macromoléculaires glissent les unes par rapport aux autres et le polymère se ramollit. Cette plage de températures est appelée température de transition vitreuse, notée T_g . A une température plus élevée, le polymère passe par un plateau caoutchoutique: son comportement est viscoélastique. A cet état, les forces de Van der Waals et la réticulation entre chaînes servent de force de rappel lorsque l'on déforme le matériau. Enfin, lorsque l'on élève la température de polymère peu réticulés, on peut assister à une phase d'écoulement visqueux, correspondant au désenchevêtrement des chaînes. Ce dernier comportement est utilisé pour mettre en forme les matières plastiques.

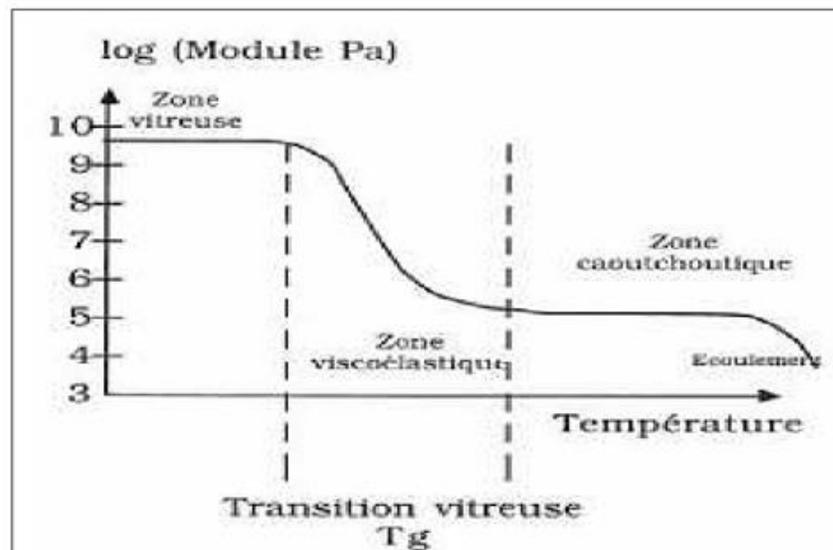


Figure I.1 : Evolution du module d'élasticité d'un polymère amorphe en fonction de la température : cas de macromolécule monodimensionnelle

Les polymères étant utilisés la plupart du temps à température ambiante, on dit qu'ils sont élastomères si leur température de transition vitreuse est inférieure à la température ambiante (comportement caoutchoutique) ils sont plastomères (comportement rigide) dans l'autre cas. [12]

I .4.3. Propriétés mécaniques :

Les propriétés mécaniques décrivent le comportement des matériaux soumis à des sollicitations mécaniques telles que des pressions, des étirements, des torsions, des frottements, des cisaillements, des chocs ou sous l'effet de la pesanteur (Figure I.2). Elles dépendent fortement du type de contrainte appliquée.

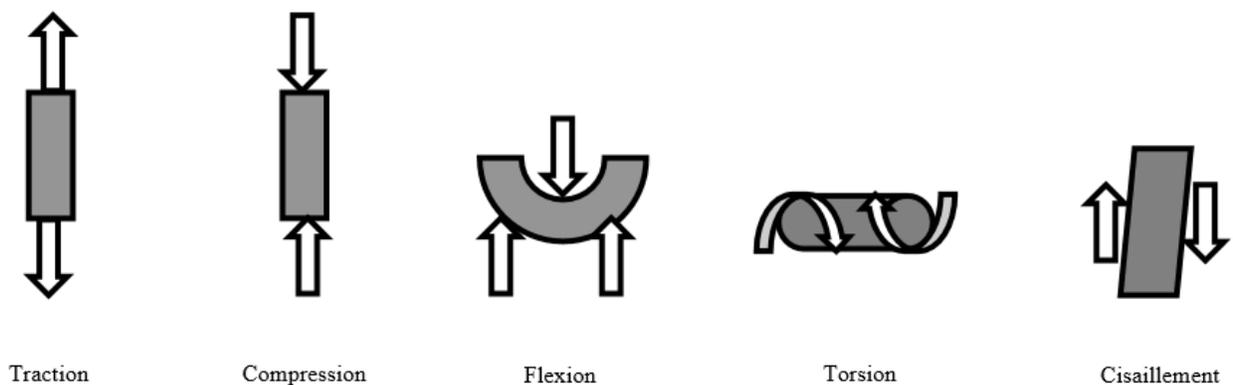


Figure I.2 : Différentes sollicitations mécaniques.

C'est la structure du réseau résultant de l'enchevêtrement des chaînes qui confère aux polymères leurs propriétés mécaniques les plus importantes, à savoir l'élasticité caoutchoutique et le comportement viscoélastique à l'état fondu d'une part, et un comportement à la fois vitreux et ductile à l'état solide d'autre part.

La Figure 1.3 présente un exemple de comportement d'un matériau soumis à une contrainte de traction. On y distingue trois zones :

- La zone 1 de la courbe contrainte – élongation est linéaire et correspond à une déformation élastique. Le matériau s'allonge sous l'action de la traction et reprend sa forme et ses dimensions initiales lorsque la contrainte est interrompue. Le coefficient de proportionnalité de la droite est appelé module de Young ou module d'élasticité. Celui-ci

est d'environ 10⁻³ GPa pour un caoutchouc, contre quelques unités pour le polystyrène, quelques dizaines pour les métaux et 10 000 pour le diamant.

- La zone 2 est appelée zone de déformation plastique. La déformation est toujours élastique, mais la variation contrainte-déformation n'est plus linéaire. Cette zone présente un maximum au-delà duquel le matériau subit une déformation irréversible. A l'arrêt de la contrainte, le matériau ne retrouve pas sa forme initiale : il existe une déformation résiduelle permanente.
- Au-delà d'une déformation critique, une contraction du matériau s'opère en un point donné jusqu'à rupture. On aborde alors la zone 3 de la courbe, dite zone de rupture. [13]

Si un matériau se casse dans son domaine élastique, c'est-à-dire sans subir de déformation plastique avant sa rupture, il est considéré comme fragile. Par contre, il sera considéré comme ductile s'il peut subir de grandes déformations plastiques avant sa rupture.

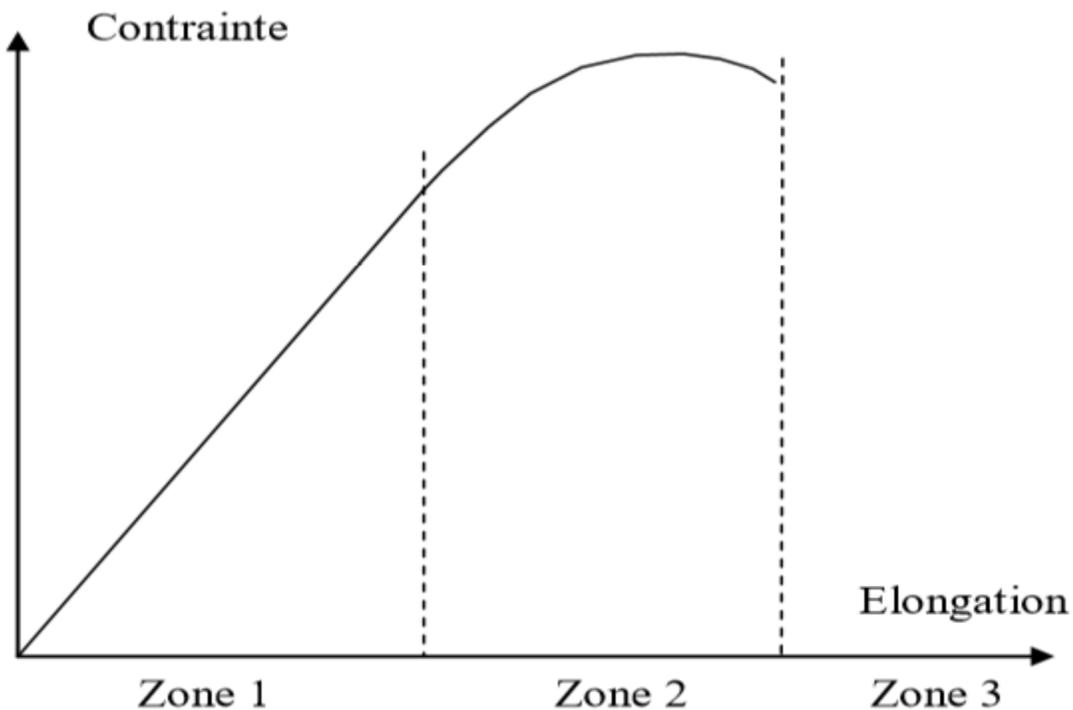


Figure I.3 : Comportement d'un matériau soumis à une contrainte mécanique.

Différents paramètres peuvent influencer les propriétés mécaniques des polymères :

- La température : il est à noter que lorsque la température de travail est inférieure à la température de transition vitreuse du matériau, la partie de la courbe à droite du maximum est courte, voire inexistante. Les chaînes se déforment peu. Au-dessus de la température de transition vitreuse, le mouvement des chaînes est possible dans les zones amorphes du matériau. Les déformations sont plus aisées. Le polymère perd de sa ténacité et le module d'élasticité diminue fortement
- La cristallinité : dans un polymère semi-cristallin, les cristallites jouent le rôle de renfort. La valeur du module d'Young est alors plus élevée que celle d'un polymère amorphe.
- La masse molaire : une valeur élevée de celle-ci signifie une longueur importante des chaînes, ce qui favorise leurs enchevêtrements dans les zones amorphes et de meilleures liaisons entre zones cristallines. Il existe ainsi une valeur seuil de la masse molaire à partir de laquelle la résistance à la rupture augmente fortement. [14]

I.5. Généralités sur le vieillissement des polymères semi cristallins :

Le vieillissement des thermoplastiques se manifeste par une dégradation souvent irréversible de leurs propriétés physiques, chimiques ou mécaniques. Ce phénomène de dégradation entraîne la notion de « durée de vie » du matériau, c'est-à-dire le temps nécessaire pour qu'une propriété atteigne un seuil critique en dessous duquel le matériau n'a plus les propriétés attendues.

Les différents mécanismes responsables de la dégradation des polymères peuvent être classés dans deux grandes catégories : les vieillissements physiques et chimiques. [15]

I.6. Déformation de la phase amorphe :

La phase amorphe du matériau est caractérisée par une partie libre contenant des enchevêtrements et une partie liée aux lamelles cristallines. Le rôle mécanique de cette phase est la transmission de contrainte d'une cristallite à une autre par l'intermédiaire de molécules de liens. Ces connexions interlamellaires sont des points de réticulation physique qui confèrent à la phase amorphe une certaine résistance mécanique.

Deux mécanismes de déformation de la phase amorphe impliquant les molécules de lien et les empilements de lamelles peuvent être rencontrés : le glissement interlamellaire et la séparation interlamellaire (Figure I-20) [16, 17].

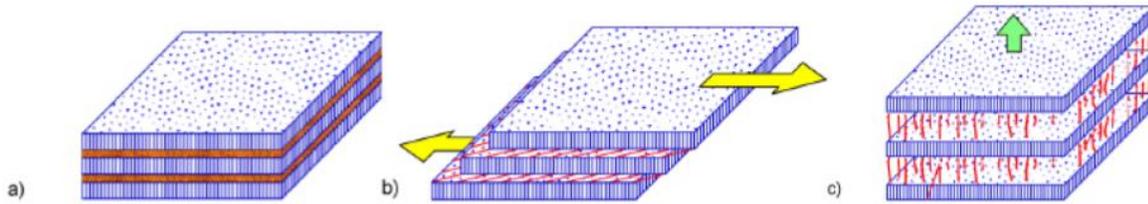


Figure I.4: Mécanisme de déformation dans la phase amorphe interlamellaire : (a) état non déformé, (b) glissement interlamellaire et (c) séparation interlamellaire.

I.7. Les facteurs affectant les polymères :

Parmi les facteurs que peuvent affectant un polymère :

I.7.1. La teneur relative en eau :

La teneur relative en eau est certainement un facteur limitant de la croissance des micro-organismes et ainsi du taux de dégradation d'un polymère dans n'importe quel milieu que ce soit. C'est le taux d'humidité relative qui va faciliter ou non l'accessibilité des micro-organismes du milieu au substrat à dégrader. [18]

I.7.2. La température :

Les températures élevées sont, en général, un facteur qui soutient la totalité des réactions abiotiques comme l'oxydation des films de polyéthylène [19]. Lorsque la température du milieu est supérieure à la température de transition vitreuse du polymère, et du fait de la mobilité des chaînes, l'absorption d'eau favorise ainsi l'hydrolyse chimique. Cela a pour conséquence d'accélérer le rendement final du processus de biodégradation. Le taux de dégradation de PLA abiotique augmente avec la température. Des tests de biodégradation effectués à 60°C sur ce polymère montrent que l'on observe une diminution de la masse molaire mais pratiquement aucune perte de masse sur les échantillons placés à 40°C et 50°C en présence de micro-organismes. Au contraire,

à 60°C, température plus élevée que la température de transition vitreuse du matériau (environ 58°C), on remarque une perte de masse significative. Il faut signaler que les petits oligomères, libérés par l'hydrolyse due à l'absorption de l'eau dans le matériau, peuvent migrer vers le milieu extérieur du fait de l'état caoutchoutique du matériau. Les mêmes résultats ont été observés par d'autres auteurs [20] dans des environnements biotiques pour des essais à des températures de 28, 40 et 55°C pour plusieurs types de polymères biodégradables [21].

I .7.3. Les rayonnements ultra-violets (UV) :

Le spectre de la lumière solaire arrivant à la surface de la terre est continu dans l'intervalle de longueurs d'onde comprises entre 290 et 1400 nm avec un maximum dans le visible autour de 500. Les rayonnements UV de longueurs d'onde inférieures à 175 nm émis par le soleil sont absorbés et réfléchis par l'oxygène alors que les rayonnements compris entre 175 et 290 nm sont absorbés par l'ozone. Les nuages ont également un rôle de filtre vis à vis du rayonnement infrarouge en ne laissant passer que les longueurs d'onde inférieures à 1400 nm. L'énergie électromagnétique émise par le soleil et qui peut atteindre la surface terrestre peut être divisée approximativement en 5% UV (290-400nm), 45% visible (400- 760nm) et 50% infrarouge (760-2400nm).

Les UV qui ne représentent qu'une faible partie du rayonnement solaire s'avèrent plus destructifs que les rayonnements visibles et infrarouges du fait de leur plus grande énergie.

La Commission Internationale de l'Eclairage (CIE) définit trois zones spectrales dans le domaine ultraviolet : UV-A, UV-B, UV-C.

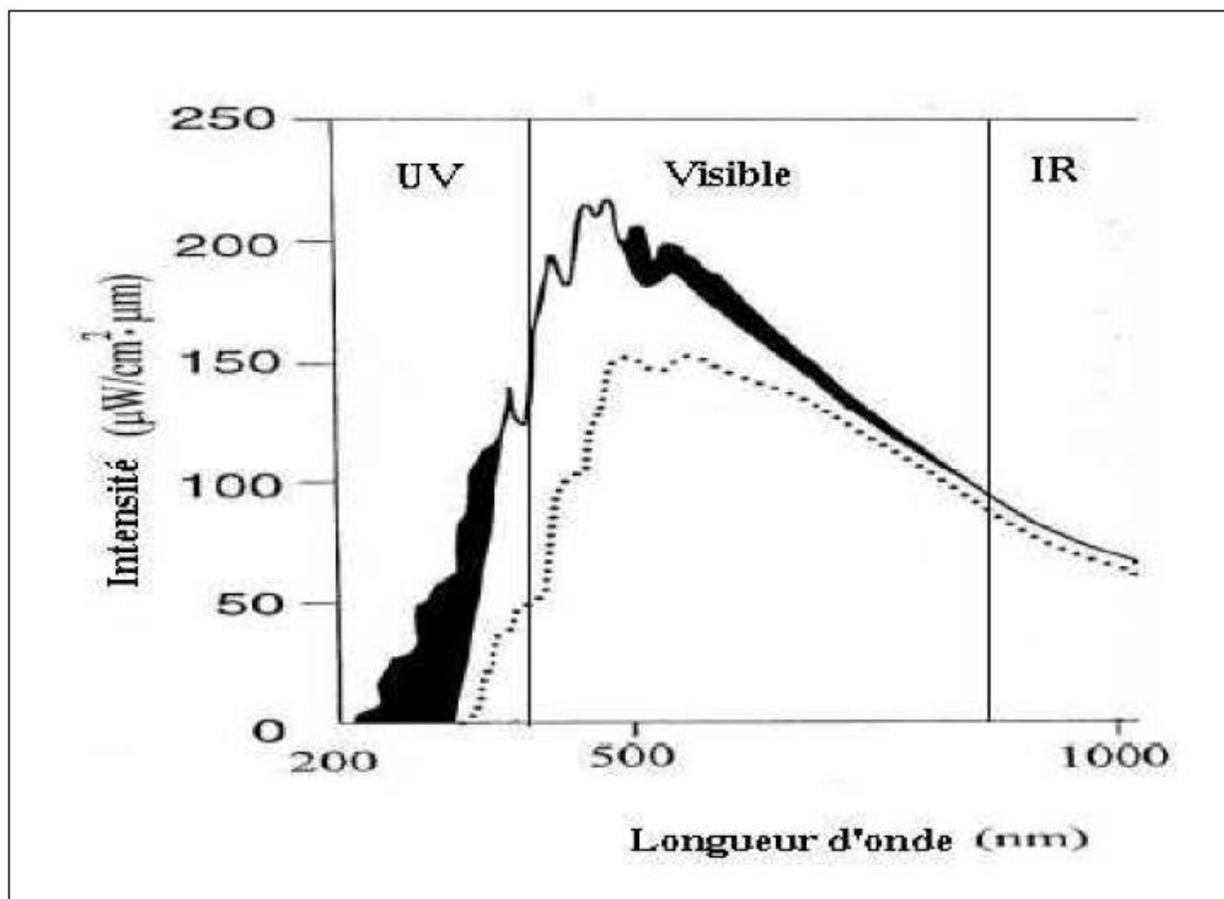


Figure I.5 : Le spectre d'émission solaire. (La surface noire est filtrée par la couche d'ozone et la ligne pointillée représente l'intensité de lumière distribuée à la surface terrestre)

La plupart des processus photochimiques conduisant à la photo dégradation des matériaux ont pour origine l'absorption des photons du domaine proche UV situé entre 290 et 400 nm. Cette zone spectrale est soumise à de fortes variations d'intensité selon la latitude, l'altitude, la pollution atmosphérique, la saison. Ainsi en hiver, l'intensité du rayonnement UV est plus faible qu'en été. Si le rayonnement UV est responsable de la plupart des altérations de surface, la lumière visible apporte également sa contribution à la photo dégradation des matériaux colorés. [22]

I .7.4. Le pH :

Pour maintenir la survie de la bactérie dans le milieu, il est nécessaire que le pH soit proche de la neutralité, ceci favorise évidemment la rupture des liaisons ester par hydrolyse et c'est ce qui a été montré avec le Poly (acide lactique). Ce dernier présente une meilleure dégradation hydrolytique

dans un milieu tamponné à pH 7,4 plutôt qu'à 3,7 [23], cela serait dû à une meilleure absorption de l'eau dans le cœur du matériau dans le cas du milieu tamponné à pH 7,4.

I.8. Présentation du Polyéthylène à basse densité (PEBD) :

Les nombreux procédés de synthèse du polyéthylène développés au cours de ces soixante dernières années conduisent à une grande plage de matériaux ayant des propriétés bien spécifiques. L'architecture chimique obtenue contrôle directement les propriétés rhéologiques et physiques.

I.9. Structure du polyéthylène :

Polyéthylène est un nom générique utilisé afin de décrire les polyoléfines issues de la polymérisation de l'éthylène. Ce sont probablement les polymères les plus couramment utilisés, en effet ils sont employés pour la fabrication des sacs plastiques, des bouteilles d'emballage et de certains jouets. Il en existe de nombreux types mais ceux-ci sont généralement classés dans deux grandes familles qui se définissent en fonction de leur masse volumique :

- Polyéthylènes basse densité ou PE-BD : $0,92 \text{ g/cm}^3 < \rho < 0,94 \text{ g/cm}^3$.
- Polyéthylènes haute densité ou PE-HD : $0,95 \text{ g/cm}^3 < \rho < 0,97 \text{ g/cm}^3$.

En 2008, la quantité produite de plastiques synthétiques a été de 245 millions de tonnes. Le polyéthylène à lui seul représente un quart de cette production en raison de son faible coût de fabrication et de ses bonnes propriétés physiques et mécaniques. De plus ce polymère permet une mise en forme généralement aisée telle que l'extrusion ou l'injection. Il possède également d'excellentes propriétés d'isolation électrique et de résistance aux chocs et présente une grande inertie chimique et biologique (contact alimentaire).

Le polyéthylène est un matériau thermoplastique obtenu par polymérisation de l'éthylène (C_2H_4) menant à des macromolécules composées par la répétition du motif $-(CH_2)-$ (Figure I-1).

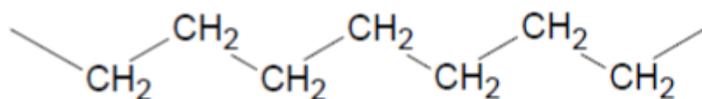


Figure I.6 : Représentation semi-développée de la chaîne macromoléculaire du polyéthylène.

C'est en 1933 dans les laboratoires de la société anglaise I.C.I., que E. Fawcett et R. Gibson découvrent un procédé de fabrication du polyéthylène fondé sur la polymérisation radicalaire de l'éthylène à haute pression conduisant à des chaînes ramifiées. Cette synthèse sera utilisée à l'échelle industrielle à partir de 1939. L'architecture macromoléculaire irrégulière résultant de cette technique a pour conséquence une cristallisation difficile du matériau qui est alors appelé polyéthylène à basse densité (PEBD) La découverte du PE-HD sous la forme d'un polyéthylène linéaire, appelé PE-L, date des années 50 et est due à quatre équipes appartenant à trois laboratoires différents. En 1945, Bailey et Reid de la Phillips Petroleum Company utilisent un catalyseur à base d'oxyde de nickel et d'oxyde de chrome pour synthétiser du PE-L. En 1950, Zletz de la Standard Oil of Indiana met au point un catalyseur à base d'oxyde de molybdène. En 1951, Hogan et Banks de la Phillips Petroleum Company améliorent le procédé existant par l'utilisation d'oxyde de chrome et d'oxyde d'aluminium. Enfin en 1953, Karl Ziegler, à l'Institut Max Planck, met au point un procédé basse pression utilisant un catalyseur appartenant à la famille de catalyseurs dits de Ziegler-Natta. De nouveaux catalyseurs furent développés dans les années 1990 par Dow et Exxon, ce sont les métallocènes, qui ont la particularité d'assurer un meilleur contrôle de la dispersion des masses moléculaires, de la dispersion des comonomères le long de la chaîne et du degré de branchement long et branchement court du polyéthylène. Aujourd'hui le PE-HD est obtenu par polymérisation cationique catalysée de l'éthylène. On utilise principalement les catalyseurs Ziegler-Natta et les catalyseurs au chrome. Les catalyseurs "Ziegler- Natta" (exemple le tétrachlorure de titane TiCl₄) sont constitués d'un composé halogéné d'un métal de transition des groupes 4 ou 5 (titane, vanadium...) et d'un composé alkylé d'un métal des groupes 2, 12, 13 (béryllium, magnésium, zinc, aluminium...). Leur productivité est supérieure à 30 kg de polymère pour 1 g de catalyseur. Les catalyseurs au chrome sont des dépôts supportés d'oxyde de chrome, réduits et activés à haute température (400 à 800°C). En 2002, la production a été réalisée à 52 % en suspension, 38 % en phase gazeuse et 10 % en solution. En 2003 la production de polyéthylène haute densité s'élevait à 505000 tonnes uniquement pour la France.[15]

I .10. Structure moléculaire (échelle I):

Une macromolécule est constituée d'une chaîne principale appelée squelette sur laquelle des atomes de nature chimique différente ou des molécules peuvent se substituer de façon plus ou moins régulière, à certains atomes appartenant au monomère constitutif d'origine. Dans le cas où les groupes d'atomes disposés latéralement sur le squelette ont une structure chimique identique à celle de la chaîne principale, on parle de ramifications ou de branches. [15] La présence de ces substituants peut avoir deux effets, soit ils rendent la chaîne principale et l'ensemble des macromolécules plus mobiles et flexibles, soit au contraire ils tendent à rigidifier l'édifice macromoléculaire. La macromolécule du polyéthylène est obtenue par la répétition de l'unité constitutive de masse molaire M_0 (Figure I-7). Le nombre de répétition n de cette unité est appelé degré de polymérisation (DP). Il est donc possible de calculer la masse molaire de la macromolécule selon l'expression : $M=n.M_0$.

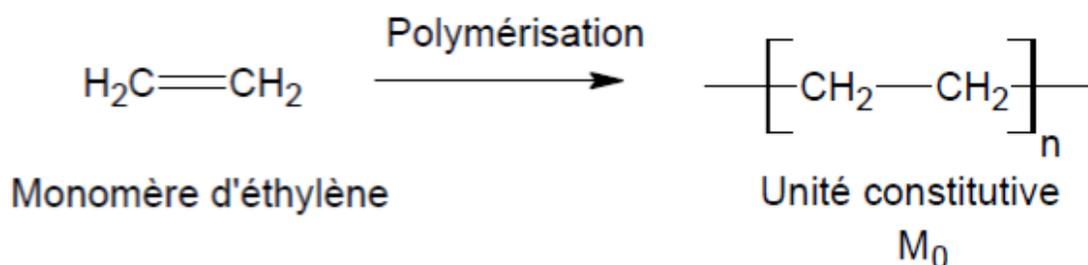


Figure I.7 : Représentation de la polymérisation du polyéthylène.

I .11. Conclusion :

Dans ce chapitre on a présenté la dégradation des matériaux polymères tels que les polyéthylènes. Cet axe a connu un développement prodigieux suite à des recherches accélérées au cours de ces dernières années.

Le champ d'utilisation des polyéthylènes est devenu de plus en plus vaste. Ceci, est dû aux diverses propriétés acquises par ce polymère et surtout à ses excellentes propriétés : dureté, souplesse, résistance du film à l'usure et aux fluides, stabilité chimique et sa bonne résistance aux rayonnements ultraviolets, à la température et à l'humidité.

Parmi les différents domaines qui nécessitent cette utilisation, on trouve le domaine de l'industrie agricole surtout pour la couverture de serre, et pour la protection, mais dans un souci de durabilité des propriétés du polyéthylène dans les conditions rudes de l'environnement, il est nécessaire de connaître au préalable le comportement à long terme de ce genre de produit. C'est pourquoi, nous cherchons dans cette présente étude, à relier l'influence des conditions climatiques (rayonnement UV, humidité, température, ph) sur la structure et le comportement des polyéthylènes basse densité et de remontre par la suite au comportement à long terme de ce type de polymère.

Chapitre2 :

Réseaux de neurones

II.1. Introduction

Les réseaux de neurones artificiels ont la spécificité d'imiter le fonctionnement du cerveau humain, qui se compose d'un nombre énorme de neurones. Ces derniers sont organisés dans un immense réseau, qui apparaît simple mais dont le fonctionnement est extrêmement compliqué. Le début de recherches sur les réseaux de neurones artificiels remonte aux années 1940 par les travaux de W.S.McCulloch et W.Pitts.

Les réseaux de neurones constituent maintenant une technique de traitement de données bien comprise et maîtrisée, constatée par leurs applications dans plusieurs domaines à savoir : l'aéronautique, l'étude spatiale, la reconnaissance de formes, la robotique, le filtrage d'informations textuelles, la classification, la régression, etc.

Toutefois, ils n'ont pas encore atteint leur plein développement, pour des raisons plus psychologiques que techniques, liées aux connotations biologiques du terme, et au fait qu'ils sont considérés, à tort, comme des outils d'Intelligence Artificielle [DRE 02].

Dans ce chapitre, nous présenterons la technique des réseaux de neurones formels d'une façon qu'elle doit être considérée comme une extension puissante de techniques mathématiques, telles que la régression.

Nous nous concentrons essentiellement sur l'utilisation de réseaux de neurones *feedforward* qui forment une classe de modèles paramétriques performante pour la régression non-linéaire.

II.2 Pourquoi les réseaux de neurones ?

On a recours au réseau de neurones lorsqu'on désire établir un modèle mathématique d'une dépendance entre les variables, et qu'on ne dispose que d'un ensemble de variables mesurées et d'un ensemble de mesures (relative à un processus de nature quelconque). En d'autres termes, on cherche un modèle mathématique qui relie des causes (entrées du modèle) à des effets (sorties du modèle).

On désigne par modèle « une boîte noire » qui constitue la forme la plus primitive de modèle mathématique : il est réalisé uniquement à partir de données expérimentales ou observations ; il peut avoir une valeur prédictive, dans un certain domaine de validité, mais il n'a aucune valeur explicative.

Ainsi, les réseaux de neurones construisent des prédicteurs à partir d'exemples. On cherche à estimer la fonction génératrice des exemples ; en d'autres termes, une fonction de régression. Cependant, le réseau de neurones MLP est le réseau le plus puissant et le plus utilisé pour la régression non linéaire [SAM 06].

II.3 Généralités

II.3.1 Neurone biologique

Nous présentons le modèle biologique juste pour comprendre l'inspiration biologique qui a été à l'origine de la première vague d'intérêt pour les neurones formels, dans les années 1940 à 1970 [MCC 43, MIN 69].

Un neurone biologique (figure II.1) est une cellule spécialisée dans le traitement et la transmission de l'information, il est constitué de trois parties : l'axone, les Dendrites et le noyau. Les dendrites forment un maillage de récepteurs nerveux qui permettent d'acheminer vers le noyau du neurone des signaux électriques en provenance d'autres neurones. Celui-ci agit comme un intégrateur en accumulant des charges électriques. Lorsque la charge accumulée dépasse un certain seuil, par un processus électrochimique, il engendre un potentiel électrique qui se propage à travers son axone vers un autre neurone. Les contacts entre deux neurones (entre axone et dendrite) se font par l'intermédiaire des synapses.

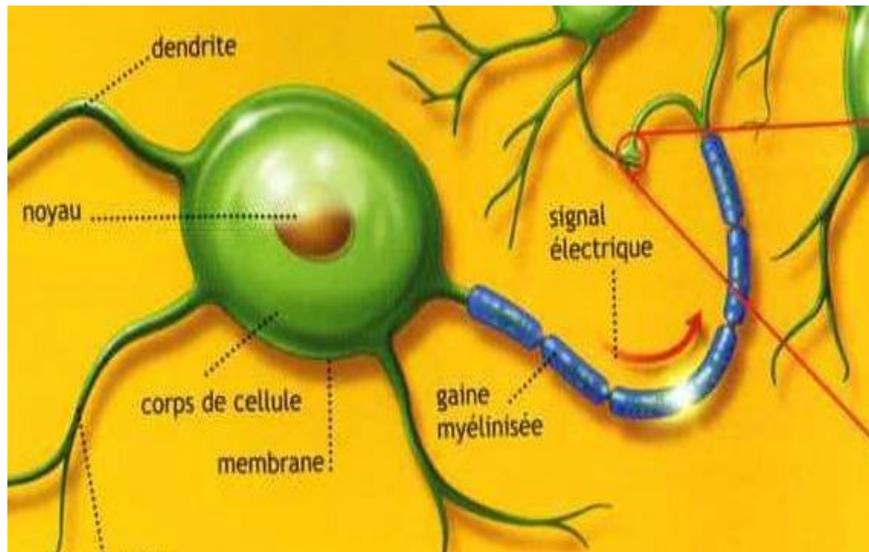


Figure II.1 : Neurone biologique.

II.3.2 Neurone formel

Le neurone formel (modèle mathématique) est illustré à la figure II.2. Un neurone est constitué d'un intégrateur qui effectue une combinaison, généralement la somme, des entrées $\{x_i\}$ pondérées par les paramètres $\{w_i\}$, qui sont souvent désignés sous le nom de « poids » (weight), ou, en raison de l'inspiration biologique des réseaux de neurones, « poids synaptiques ». La combinaison linéaire est appelée « potentiel » à laquelle s'ajoute un terme constant ou « biais » w_0 , appelé souvent le seuil d'activation du neurone :

$$y = w_0 + \sum_{i=1}^n w_i x_i$$

Le résultat y est ensuite transformé par une fonction d'activation f qui produit la sortie du neurone.

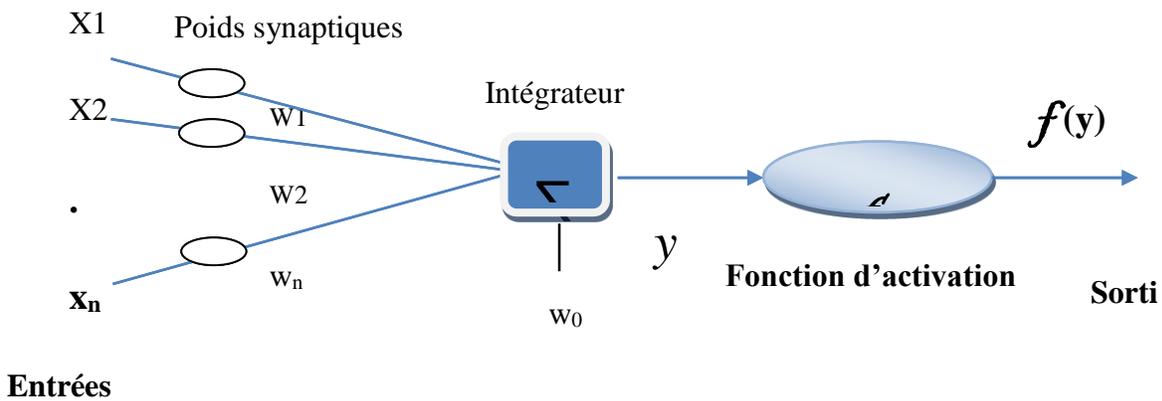


Figure II.2 : Neurone formel

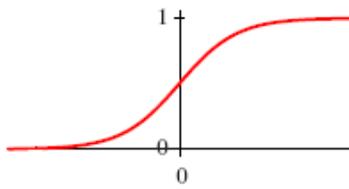
II.3.3 Fonction d'activation

La fonction d'activation joue un rôle prépondérant dans le comportement du neurone et du réseau entier. Plusieurs fonctions d'activation ont été examinées dans [DUC 99], la plus utilisée est la fonction *sigmoïde* (généralement c'est la fonction tangente hyperbolique) ; car elle introduit de la non-linéarité, et c'est aussi une fonction continue, différentiable.

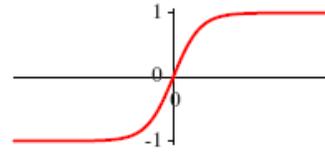
Les principales fonctions d'activation que l'on retrouve dans les différentes architectures sont représentées sur (la figure II.3).

II.3.4 Définition d'un réseau de neurones

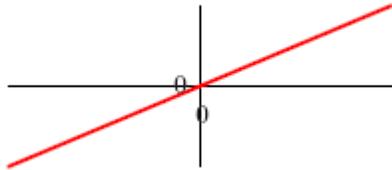
Un neurone (neurone formel) est une fonction algébrique non linéaire, paramétrée, à valeurs bornées [DRE 02]. Un neurone formel n'a d'intérêt que s'il est connecté à d'autres neurones pour former un réseau. Un tel réseau est appelé « Réseau de neurones ».



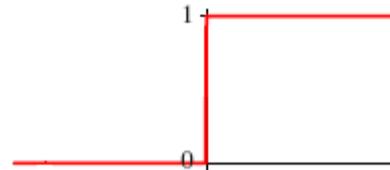
a) $f(x) = \frac{1}{1+e^{-x}}$



b) $f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$



c) $f(x) = x$



d) $f(x) = \begin{cases} 1 \\ 0 \end{cases}$

Figure II.3 : Les principales fonctions d'activation : (a) fonction logistique, (b) fonction tangente hyperbolique, (c) fonction linéaire, (d) fonction à seuil

II.4 Réseaux de neurones pour la régression non-linéaire

II.4.1 La régression non linéaire

La régression non linéaire est une des méthodes classiques de la statistique largement utilisée dans le traitement des données. Son but principal est d'aider à la détermination d'une fonction univoque permettant de relier deux variables distinctes pour lesquels on fait l'hypothèse qu'il existe une relation de dépendance fonctionnelle. On se référera au livre de Bates [BAT 88] pour plus de détails.

Considérons une grandeur mesurable y_p , dépendant d'un ensemble de facteurs mesurés qui constituent les composantes d'un vecteur x . On considère les résultats des mesures y_p comme des réalisations d'une variable aléatoire Y , et les valeurs des grandeurs d'entrée comme des réalisations d'un vecteur aléatoire X . On cherche donc à estimer l'espérance mathématique de la variable aléatoire Y pour une réalisation donnée x du vecteur aléatoire X ; on la note $\mathbf{E}_Y(x)$. Cette quantité est donc une fonction de x ; elle est appelée fonction de régression de la variable aléatoire Y .

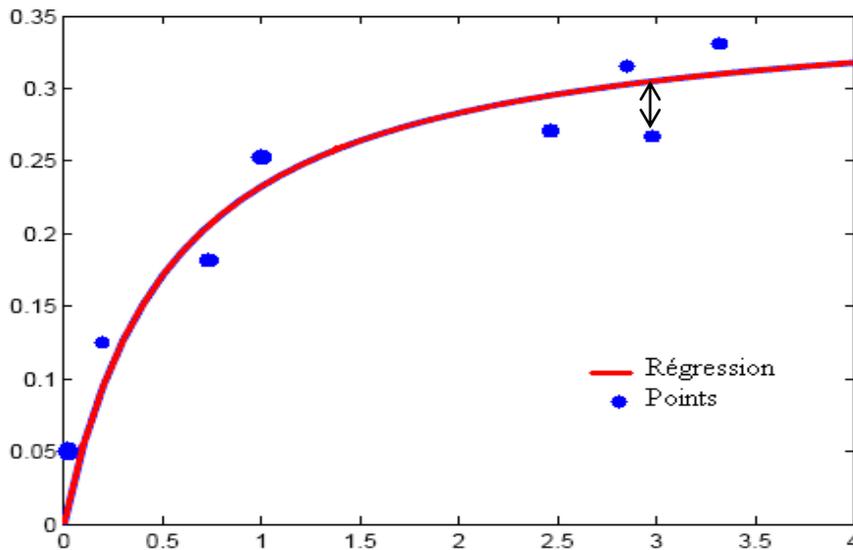


Figure II.4 : Courbe de régression non linéaire pour des données

II.4.2 Modèle non linéaire par rapport aux variables

Soit $g(x,w)$ un modèle dont le vecteur des variable est x et le vecteur des paramètres est w . s'il existe un vecteur de paramètres w_p tel que le modèle soit identique à la fonction de régression $\mathbf{E}_Y(x)$: $g(x,w_p) \equiv \mathbf{E}_Y(x)$, on dit que la famille des fonctions $g(x,w)$ contient la régression, ou encore le modèle $g(x,w)$ est vrai. Si ce n'est pas le cas, on se contentera de chercher un modèle aussi proche que possible de la fonction de régression $\mathbf{E}_Y(x)$. Or, un réseau de neurone est un modèle non linéaire par rapport aux variables, et la recherche d'un modèle proche de la fonction $\mathbf{E}_Y(x)$ n'est que l'apprentissage. Nous allons détailler ces points dans la section II.8

II.5 Architecture des réseaux de neurones

Un réseau de neurones est représenté par un graphe dont les nœuds sont les neurones et les arrêtes les connexions entre ceux-ci. Pour décrire l'architecture de réseau on emploie la matrice de poids de connexions $W = [w_{ij}]$, où le w_{ij} dénote le poids de connexion du nœud j (ou de l'entrée j) au nœud i . quand $w_{ij}=0$, il n'y a pas de connexion entre le nœud j et le nœud i .

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

II.5.1 Les réseaux de neurones non bouclés (réseau statique)

Sont des réseaux à propagation vers l'avant (*FNNs, feedforward neural networks*) [DU 06]. Le réseau est organisé en couche successives, l'information est ainsi transmise de manière unidirectionnelle du neurone j vers le neurone i . la connexion entre deux couches successives peut être totale ou partielle (figure II.5).

Les données du problème (les entrées) sont présentées à la première couche cachée, la dernière couche est la couche de sortie représentant le résultat. Ainsi, les sorties peuvent être exprimées comme des fonctions déterministes des valeurs d'entrées.

Il n'existe aucune méthode optimale pour déterminer a priori la structure interne d'un réseau de neurones. Elle ne peut être déterminée que de façon expérimentale.

Les réseaux de neurones non bouclés sont souvent appelés *réseau statique* et cela au fait que : le temps ne joue aucun rôle fonctionnel dans un réseau de neurones non bouclé : si les entrées sont constantes, les sorties le sont également. Le temps nécessaire pour le calcul de fonction réalisée par chaque neurone est négligeable, et fonctionnellement, on peut considérer ce calcul comme instantané.

II.5.2 Les réseaux de neurones bouclés (réseau dynamique)

Le deuxième type de réseaux de neurones est les réseaux de neurones bouclés ou récurrents (*RNNs, Recurrent Neural Networks*), dont l'architecture est plus générale, le graphe des connexions est *cyclique* : lorsqu'on se déplace dans le réseau en suivant le sens des connexions, il est possible de trouver au moins un chemin qui revient à son point de départ. La fonction de neurone peut être d'elle-même ; pour cela, la notion de *temps* est explicitement prise en considération.

Ainsi, à chaque connexion d'un réseau de neurones bouclé est attaché outre que le poids, un *retard*, multiple entier de l'unité de temps choisis.

Un réseau de neurones bouclé à temps discret réalise une (ou plusieurs) *équations aux différences non linéaires*, par composition des fonctions réalisées par chacun des neurones et des retards associés à chacune des connexions [DRE 02].

La forme la plus générale des équations régissant un réseau de neurones bouclé est appelée forme canonique [NER 93] :

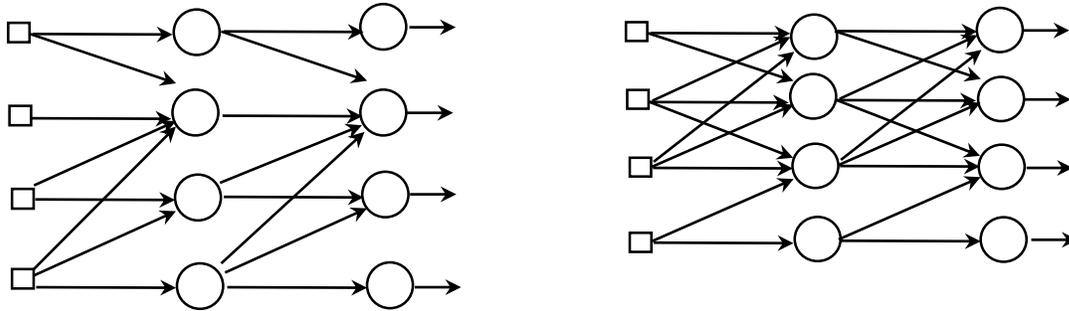
$$x(k+1) = \varphi[x(k), u(k)]$$

$$y(k) = \psi[x(k), u(k)]$$

Où φ et ψ sont des fonctions non linéaires réalisées par un réseau de neurones non bouclé, k désigne le temps (discret).

Le modèle de Hopfield [HOP 82] et la machine de Boltzman [ACK 85] sont les réseaux bouclés les plus populaires.

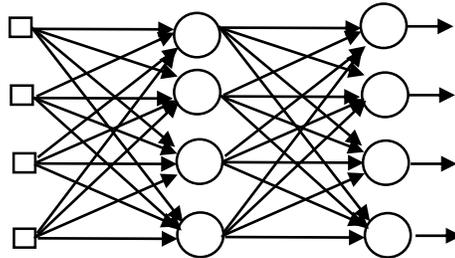
On peut se référer au livre de *K-L.Du* [DU 06] pour une présentation plus complète des réseaux de neurones bouclés.



Entrées Couches cachées Couches de sortie

b) Connexions locales

a) Connexions directes



c) Connexions totales

Figure II.5 Réseaux de neurones feedforward [PAL 06].

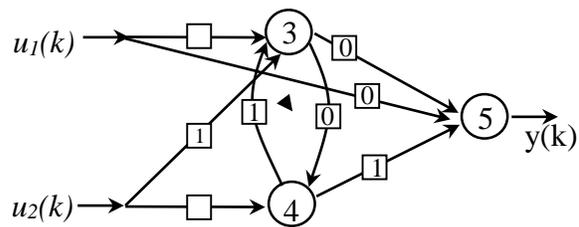


Figure II.6 : Exemple d'un réseau de neurones bouclé. Les nombres dans les carrés sont les retards (exprimés en nombre de périodes d'échantillonnage) associés à chaque connexion [DRE 02].

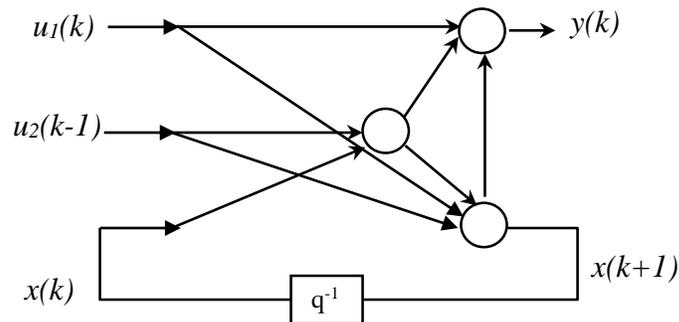
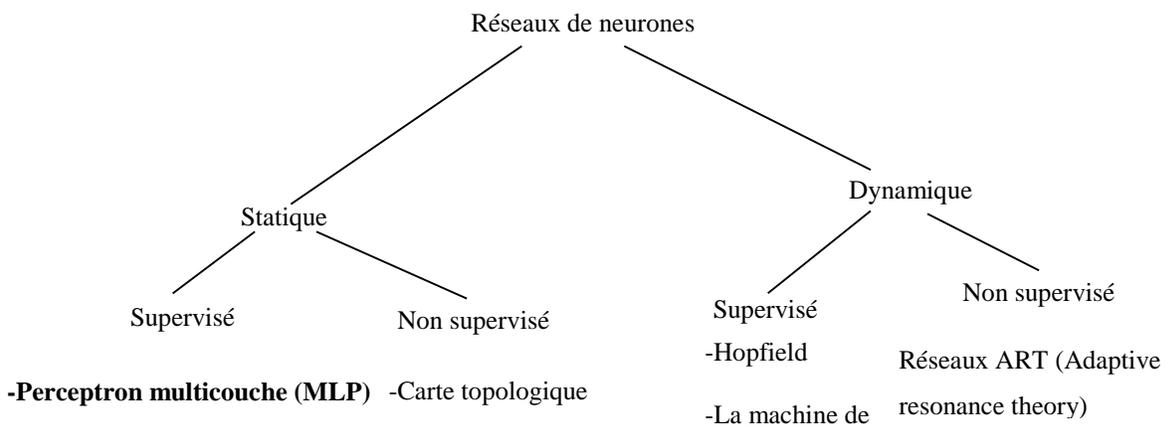


Figure II.7 : La forme canonique du réseau de la Figure II.6 [DRE 02].

III.5.3 Taxonomie des réseaux de neurones (non exhaustive)



II.6 Le perceptron multicouche MLP

Le perceptron multicouche (MLP, Multi Layer Perceptron) est un réseau de type FNNs (voir section II.5.1). Le réseau est organisé en couche successive. Un neurone ne peut être connecté qu'à la totalité des neurones d'une couche limitrophe. Ainsi il n'y a pas de connexions entre les neurones d'une même couche, et l'information circule des entrées vers les sorties sans retour en arrière (figure II.8).

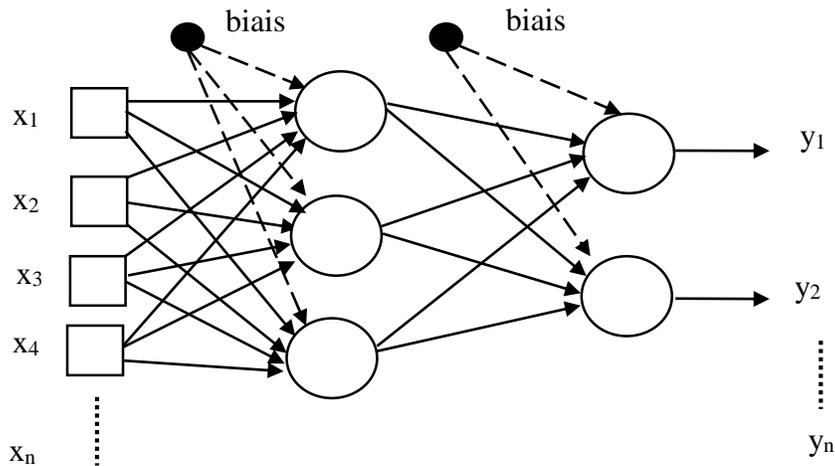


Figure II.8 : réseaux MLP à une couche cachée

Le graphe d'un réseau de neurones non bouclé est acyclique (figure II.8) : si l'on se déplace dans le réseau, à partir d'un neurone quelconque, en suivant les connexions on ne peut pas revenir au neurone de départ.

Les entrées (représentées par des carrés sur la figure II.8) ne sont pas des neurones : elles ne réalisent aucun traitement de l'information, puisqu'elles ne font que transmettre les valeurs des variables. (On voit souvent des textes qui mentionnent les entrées comme couche d'entrée ou neurone d'entrée, alors que cette expression est trompeuse).

Formellement, soit X un vecteur d'entrée tel que $X = (x_1, x_2, \dots, x_n)$, la propagation avant permet de récupérer sur la couche de sortie un vecteur $Y = (y_1, y_2, \dots, y_m)$. Donc, un MLP définit une fonction de \mathcal{R}^n dans \mathcal{R}^m .

Pour un MLP d'architecture *fixé*, la fonction définie par le réseau dépend des valeurs des poids W de ses différentes connexions. Une architecture génère donc une famille de fonctions :

$$\begin{aligned} \mathcal{R}^n &\longrightarrow \mathcal{R}^m \\ X &\longrightarrow Y = \mathbf{g}(X, W) \end{aligned}$$

L'ensemble des familles définies par l'ensemble des architectures possibles est très adapté à la recherche de fonctions de régression non-linéaires.

II.6.1 Propriétés d'approximations du MLP

La propriété fondamentale des réseaux MLP est qu'ils sont des approximateurs universels. En effet, Il a été mathématiquement démontré qu'un réseau MLP composé d'une seule couche cachée avec pour fonction d'activation une fonction sigmoïde et d'une couche de sortie linéaire, peut approcher bien toute fonction multi-variable continue avec une précision arbitraire [CYB 89, FUN 89, HOR 89, XIA 05].

C'est ce qui justifie l'utilisation *réelle* des réseaux MLP : la recherche d'une fonction de régression à partir d'un nombre *fini* de points.

Par ailleurs, un réseau à deux couches cachées peut mieux rapprocher toute fonction continue, mais ceci va présenter des minimums locaux supplémentaires [CHE 90, XIA 05].

II.7 Les réseaux RBF

Les réseaux de neurones à fonctions radiales de base (*Radial Basis Function Network*) sont également des réseaux à propagation avant (*feedforwd*). Ils possèdent une seule couche cachée composée de fonctions noyaux (φ) généralement gaussiennes. Les paramètres de cette dernière sont attachés à la non-linéarité du neurone, et non pas aux entrées du neurone.

Deux paramètres caractérisent ces fonctions noyaux [DU 06] :

- Un vecteur de référence μ_j appelé centre ou prototype ;
- La dimension σ_i du champ d'influence appelé rayon d'influence.

Chaque neurone de la couche cachée est caractérisé par la distance du vecteur d'entrée x au vecteur prototype μ_i : $r_i(x) = \|x - \mu_i\|$.

Donc, La fonction gaussienne : $\varphi_i(x) = \exp\left(\frac{-r_i(x)^2}{\sigma_i^2}\right)$

Chaque neurone de sortie (linéaire) réalise une somme pondérée des sorties des neurones cachés.

La différence pratique entre les MLP et les RBF est la suivante : 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 (MLP).

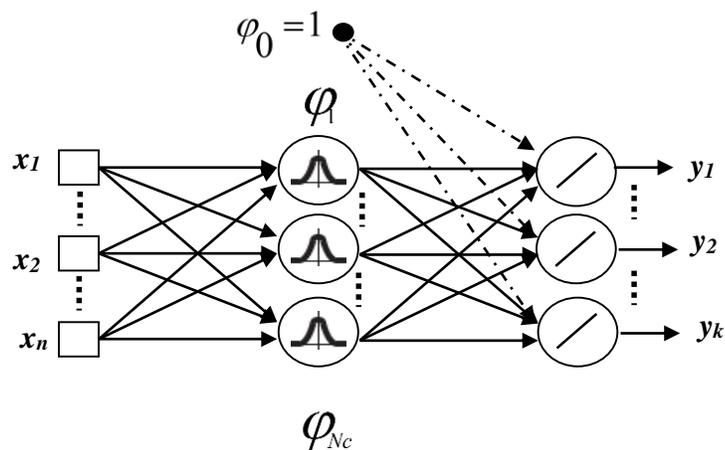


Figure II.9 : Réseau RBF.

$\varphi_0(x) = 1$ Correspond au biais de la couche de sortie

II.8 L'apprentissage des réseaux de neurones

On appelle « apprentissage » des réseaux de neurones la procédure qui consiste à estimer les paramètres des neurones du réseau, afin que celui-ci remplisse au mieux la tâche qui lui est affectée [DRE 02].

On entend par paramètres les poids synaptiques (w_{ij}), qui sont ajustés plusieurs fois selon un processus graduel, itératif avant d'atteindre leurs valeurs finales. Cet ajustement a pour but de

minimiser l'erreur en sortie. Ce processus itératif s'arrêtera soit en trouvant une solution optimale, soit en atteignant un nombre d'itérations fixé à l'avance.

On peut distinguer deux types d'apprentissages : l'apprentissage supervisé et l'apprentissage non supervisé.

III.8.1 L'apprentissage supervisé

Un réseau de neurone non bouclé réalise une relation non-linéaire $Y=g(X, W)$ comme modèle de régression. On cherche à estimer les paramètres d'un réseau de neurones qui réalise une approximation de la fonction de régression, celle-ci est inconnue.

On connaît, en tous points ou seulement en certains points, les valeurs que doit avoir la sortie (Y) du réseau en fonction des entrées(X) correspondantes : c'est en ce sens que l'apprentissage est *supervisé*.

Donc, on cherche la fonction de régression ($E(y/x)$: espérance mathématique des valeurs observées y au point x) et, comme le nombre de point est fini, on ne peut y trouver qu'une approximation.

Pour trouver cette approximation, il faut définir une fonction de coût qui mesure l'écart entre la sortie du modèle (fonction réalisée par le réseau de neurone) et la sortie désirée. Plus la valeur de la fonction de coût est petite, plus le modèle reproduit fidèlement les observations utilisées pour l'apprentissage. L'algorithme d'apprentissage cherche donc à trouver le point, dans l'espace des paramètres, pour lequel la fonction de coût est minimale.

L'apprentissage des réseaux de neurones non bouclé se ramène donc au problème de la minimisation d'une fonction de coût (i.e. *Un problème d'optimisation sans contraintes*).

La procédure d'apprentissage supervisé nécessite :

1. *Un ensemble d'exemple d'apprentissage ;*
2. *La définition d'une fonction de coût qui mesure l'écart entre les sorties du réseau de neurones et les sorties désirées ;*

3. Un algorithme de minimisation de la fonction de coût par rapport aux paramètres.

II.8.1.1 L'ensemble d'exemple d'apprentissage

C'est un ensemble de couples d'observations $\{(entrée, sortie)\}$, ces couples d'observations (x_i, y_i) , $i=1 \dots N$, sont souvent des représentations bruitées ou ambiguës d'une réalité sous-jacente.

Le nombre d'exemples de l'ensemble d'apprentissage a une grande importance.

En effet, comme la sortie du réseau est non linéaire par rapport aux paramètres, la fonction de coût peut présenter des minima locaux, et les algorithmes d'apprentissage ne donnent aucune garantie de trouver le minimum global [DU 06]. Toutefois, si le nombre d'exemples est insuffisant, non seulement des minima locaux apparaissent, mais, le minimum global de la fonction de coût ne correspond pas forcément aux valeurs des paramètres recherchés. Par contre si l'on dispose d'un nombre suffisant d'exemples, le problème des minima locaux ne se pose pratiquement pas : il suffit, d'effectuer quelques apprentissages avec des initialisations différentes des paramètres.

II.8.1.2 Normalisation des entrées et des sorties

Avant tout apprentissage, il est indispensable de normaliser et de centrer toutes les variables d'entrées. En effet, si des entrées ont des grandeurs très différentes, celles qui sont « petites » n'ont pas d'influence sur l'apprentissage. Il est donc recommandé, pour chaque vecteur d'entrée $X_i = (x_1, x_2, \dots, x_n)$, de calculer la moyenne μ_i et l'écart type σ_i de ses composantes, et d'effectuer le changement de variable $x'_i = (x_i - \mu_i) / \sigma_i$ [DRE 02].

La normalisation des sorties $Y_i = (y_1, y_2, \dots, y_m)$ s'effectue avec la même technique.

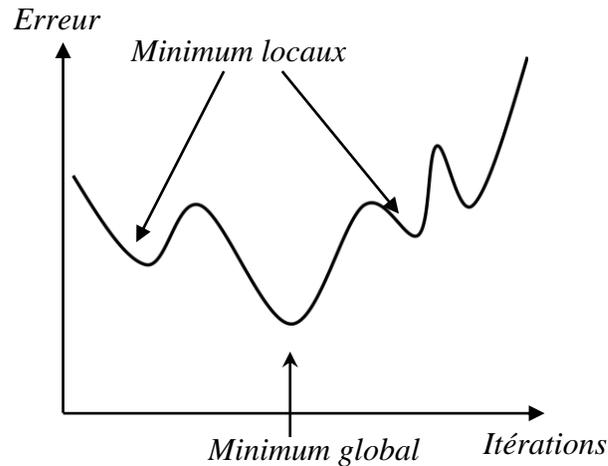


Figure II.10 : Exemple de minima locaux

II.8.1.3 Fonction de coût

La fonction de coût doit permettre de mesurer l'écart entre le modèle et les observations. Il existe un grand nombre de fonction [DU 06] ; nous présenterons la fonction du coût des *moindres carrés*, qui est la plus utilisée.

Pour un exemple i d'un ensemble d'observation E , la fonction de coût des moindres carrés est égale à la somme, sur les N_s neurones de la couche de sortie, des carrés des écarts entre la sortie du modèle (sortie du réseau de neurone = S^i) et la sortie désirée d^i . Comme la sortie du réseau dépend du vecteur de paramètre W , la fonction de coût en dépend également. On la note $J^i(w)$:

$$J^i(w) = \sum_{q=1}^{N_s} (d_q^i - S_q^i)^2 \quad (2.1)$$

Sur un ensemble d'exemple M la fonction de coût est notée $J^M(w)$ et elle est définie par la moyenne des carrés des écarts sur les N exemples de cet ensemble (Ecart Quadratique Moyen).

$$J^M(w) = \frac{1}{N} \sum_{i=1}^N J^i(w) \quad (2.2)$$

II.8.1.4 Algorithme d'optimisation

A partir d'une fonction de coût dépendant du vecteur de paramètres (w) et des exemples de l'ensemble d'apprentissage, il faut choisir l'algorithme d'optimisation qui permettra d'estimer le vecteur des paramètres pour lequel la fonction de coût choisie est minimale. De nombreux algorithmes ont été proposés : la rétropropagation de l'erreur, gradients conjugués, la méthode de levenberg-Marquardt, les méthodes par région de confiance, les filtres de kalman... [DU 06]. Nous n'aborderons ici que les méthodes les plus classiques.

Le principe de ces méthodes est de se placer en un point, de trouver une direction de descente du coût dans l'espace des paramètres (w), et ensuite, de se déplacer d'un *pas* suivant cette direction. On atteint un nouveau point et l'on recommence la procédure. On poursuit cette démarche jusqu'à satisfaction d'un critère d'arrêt.

Les méthodes d'optimisation non linéaires que nous présentons se différencient par le choix de la *direction de descente* et du *pas*.

Deux types de méthodes peuvent être mis en œuvre pour minimiser la fonction de coût :

- Des méthodes non adaptatives d'apprentissage (*batch training*) : elles consistent à estimer les paramètres du modèle par minimisation de la fonction de coût des moindres carrés, qui tient compte simultanément de tous les exemples de l'ensemble d'apprentissage ; l'utilisation d'une telle méthode nécessite que les N exemples soient disponibles dès le début de l'apprentissage ;
- Des méthodes adaptatives d'apprentissage (*online training*) : elles consistent à modifier les paramètres du modèle, successivement en fonction du coût partiel relatif à chaque exemple k : $j^k(w) = (d_q^k - S_q^k)^2$, car les exemples sont présentés au réseau de neurone un par un. Cette méthode est appelée souvent méthode incrémentale. Cette technique est utilisée lorsqu'on désire effectuer l'apprentissage au fur et à mesure de la disponibilité des exemples.

L'objectif est d'avoir une erreur quadratique minimum sur les points de l'ensemble d'apprentissage, et non pas une erreur nulle. Du fait que, les mesures étant entachées de bruit. Un modèle parfait reproduirait le bruit et les perturbations, alors que l'objectif de la modélisation est, bien au contraire, d'extraire tout ce qui est déterministe et prédictible dans l'information présente dans les données. Nous aborderons plus en détails dans la section III.9 les problèmes liés à la généralisation.

II.8.1.5 Calcul du gradient par rétropropagation de l'erreur

C'est l'algorithme le plus connu, établi par [RUM 86], qui permet d'utiliser les méthodes d'optimisation non-linéaire basées sur le gradient. Il est non seulement employé pour les réseaux non bouclés FNNs tel que le MLP mais, il a été également adapté aux réseaux bouclés RNNs [PIN 87].

Soit (x, y) un exemple de la base d'apprentissage. Le neurone i calcul une grandeur y_i qui est une fonction non linéaire de son potentiel v_i ; le potentiel v_i est une somme pondérée des entrées x_j , la valeur de l'entrée x_j étant pondérée par un paramètre w_{ij} :

$$y_i = f \left(\sum_{j=1}^{n_i} w_{ij} x_j \right) = f(v_i)$$

Les n_i entrées du neurone i peuvent être soit les sorties d'autres neurones, soit les entrées du réseau. Dans toute la suite, x_j désignera donc indifféremment soit la sortie y_j du neurone j , soit l'entrée j du réseau.

L'objectif est de calculer le gradient de l'équation (2.2). Or pour évaluer le gradient, il suffit d'évaluer le gradient du coût partiel $J^k(w)$ relatif à l'observation k , et de faire ensuite la somme sur tous les exemples.

L'algorithme de rétropropagation consiste essentiellement en l'application répétée de la règle de dérivées composées. On remarque tout d'abord que la fonction de coût partielle ne dépend du paramètre w_{ij} que par l'intermédiaire de la valeur de la sortie du neurone i , qui est elle-même fonction uniquement du potentiel du neurone i ; donc :

$$\left(\frac{\partial j^k}{\partial w_{ij}} \right)_k = \left(\frac{\partial j^k}{\partial v_i} \right)_k \left(\frac{\partial v_i}{\partial w_{ij}} \right)_k = \delta_i^k x_j^k \quad (2.3)$$

où :

$\left(\frac{\partial j^k}{\partial v_i} \right)_k$ Désigne la valeur du gradient du coût partiel par rapport au potentiel du neurone i

$\left(\frac{\partial v_i}{\partial w_{ij}} \right)_k$ Désigne la valeur de la dérivée partielle du potentiel du neurone i par rapport au paramètre w_{ij} lorsque les entrées du réseau sont celles qui correspondent à

x_j^k est la valeur de l'entrée j du neurone i lorsque les entrées du réseau sont celles qui correspondent à l'exemple k .

Les quantités δ_i^k sont calculées d'une manière récursive, en menant les calculs depuis les sorties du réseau vers ses entrées.

En effet :

- Pour le neurone de sortie i :

$$\delta_i^k = \left(\frac{\partial j^k}{\partial v_i} \right)_k = \left(\frac{\partial [(y_p^k - g(x^k, w))^2]}{\partial v_i} \right)_k = -2 [y_p^k - g(x^k, w)] \left(\frac{\partial g(x, w)}{\partial v_i} \right)_k$$

Or, la sortie du modèle est la sortie y_i du neurone de sortie ; cette relation s'écrit donc :

$$\delta_i^k = -2 [y_p^k - g(x^k, w)] f'(v_i^k) \quad (2.4)$$

où $f'(v_i^k)$ désigne la dérivée de la fonction d'activation du neurone de sortie lorsque les entrées du réseau sont celles de l'exemple k .

- Pour un neurone caché i :

La fonction du coût ne dépend du potentiel du neurone i que par l'intermédiaire des potentiels des neurones m qui reçoivent la valeur de la sortie du neurone i , c'est-à-dire de tous les neurones qui, dans le graphe de connexions du réseau, sont adjacents au neurone i , entre ce neurone et la sortie :

$$\delta_i^k \equiv \left(\frac{\partial j^k}{\partial v_i} \right)_k = \sum_m \left(\frac{\partial j^k}{\partial v_m} \right)_k \left(\frac{\partial v_m}{\partial v_i} \right)_k = \sum_m \left(\frac{\partial j^k}{\partial v_m} \right)_k \left(\frac{\partial v_m}{\partial v_i} \right)_k$$

Or $v_m^k = \sum_i w_{mi} x_i^k = \sum_i w_{mi} f(v_i^k)$ d'où $\left(\frac{\partial v_m}{\partial v_i} \right)_k = w_{mi} f'(v_i^k)$.

Finalement, on obtient la relation :

$$\delta_i^k = \sum_m \delta_m^k w_{mi} f'(v_i^k) = f'(v_i^k) \sum_m \delta_m^k w_{mi} \quad (2.5)$$

Donc, l'algorithme de rétropropagation comporte deux phases pour chaque exemple k :

- Une phase de propagation, au cours de laquelle les entrées correspondant à l'exemple k sont utilisées pour calculer les sorties et les potentiels de tous les neurones.
- Une phase de rétropropagation, au cours de laquelle sont calculées les quantités δ_i^k . une fois que ces quantités sont disponibles, on calcul les gradients des coûts partiels par la relation (3.3), puis il suffit de faire la somme pour obtenir le gradient de la fonction « coût total ».

II.8.1.6 Modification des paramètres

Après évalué le gradient de la fonction de coût par rapport aux paramètres du modèle, on effectue une modification des poids, afin d'approcher d'un minimum de la fonction de coût.

Nous allons citer quelques algorithmes, les plus couramment utilisées, de minimisation itérative des paramètres du modèle.

- **La descente du gradient**

La méthode du gradient consiste à modifier les paramètres par la formule suivante, à l'itération i de l'apprentissage :

$$w_i = w_{(i-1)} - \eta_i \nabla j(w_{(i-1)}) \quad \text{avec } \eta_i > 0 \quad (2.6)$$

C'est une méthode de descente itérative qui à chaque itération déplace la solution dans la direction la plus forte pente, c'est-à-dire la direction opposé au gradient. La descente du gradient converge vers un minimum local de l'erreur. La quantité η_i est appelé *pas du gradient* ou *pas* d'apprentissage.

Plusieurs heuristiques ont été proposées pour déterminer la valeur de η_i :

- Un pas d'apprentissage constant, $\eta_i = \eta_0$,
- Un pas adaptatif qui est augmenté lorsque la solution courante diminue l'erreur et réduit dans le cas contraire.

Pour accélérer la convergence de la méthode du gradient, [PLA 86] proposent d'ajouter un terme d'inertie dans la formule de mise à jour des poids :

$$\Delta w_t = -\eta_t \nabla j(w_t) + \mu \Delta w_{t-1} \quad (2.7)$$

Le scalaire μ est appelé moment (*momentum*), qui prend ses valeurs dans $[0,1]$ (typiquement $\mu=0,1$).

- **Les gradients conjugués**

La méthode des gradients conjugués [HES 52, Mol 93] consiste à choisir une direction conjuguée à la direction de la descente du gradient :

$$d_t = -\nabla j(w_{(t)}) + \beta_t d_{t-1} \quad \text{avec } d_{t-1} \text{ direction de descente du gradient.}$$

$$w_{t+1} = w_t + \eta_t d_t \quad (2.8)$$

Où le pas d'apprentissage η_t est le résultat de la recherche linéaire dans la direction d_t et où β_t est déterminé généralement par la formule de Polak-Ribière [MAH 05]:

$$\beta_t = \frac{g_t^T (g_t - g_{t-1})}{g_{t-1}^T g_{t-1}} \quad \text{où } g_t = \nabla j(w_{(t)})$$

Cette formule permet de réinitialiser d_t à la direction de plus forte pente lorsque l'algorithme ne progresse plus (lorsque $g_t - g_{t-1}$ devient négligeable).

On peut remarquer que la descente du gradient avec *momentum* est une approximation de la méthode des gradients conjugués. Or, avec la descente du gradient l'utilisateur doit choisir empiriquement les valeurs du pas d'apprentissage et du *momentum*, celles-ci sont déterminées automatiquement pendant l'apprentissage en utilisant la méthode des gradients conjugués.

- **Levenberg-Marquardt**

La méthode de Levenberg-Marquardt [LEV 44, MAR 63] consiste à modifier les paramètres selon la relation suivante :

$$w_k = w_{k-1} - [H_{k-1}(j(w_{k-1})) + \lambda_{k-1} \cdot I]^{-1} \nabla j(w_{(k-1)})$$

où I est la matrice identité qui assure l'ajout du terme de relaxation variable λ_{k-1} aux éléments de la diagonale de $H_{k-1}(j(w_{k-1}))$. A chaque itération k le facteur de relaxation λ_{k-1} est incrémenté ou décrémenté. Ainsi, λ_{k-1} diminue à l'approche de l'optimum et augmente dans le cas contraire. Typiquement $\lambda_0 = 0,1$.

II.8.1.7 Résumé de la rétropropagation du gradient

Etape 1 : initialisation des poids par des petites valeurs aléatoires.

Etape 2 : présentation d'un vecteur d'entrée et celui de la sortie désirée.

Etape 3 : calcul de façon itérative des états des neurones dans les couches suivantes

(k=1 à k) :

$$V_j^k = F\left(\sum_{i=1}^n (W_{ji}^k V_i^{k-1} + b_j^k)\right)$$

Avec :

F : la fonction d'activation

k : Nombre de couches du réseau.

n : Nombre de neurones dans la couche k-1.

V_j^k : La sortie du neurone j de la couche k.

V_j^0 : Entrée du neurone j de la couche d'entrée.

W_{ji}^k : Connexion entre le neurone i de la couche k-1 et le neurone j de la couche k.

b_j : biais de neurone j de la couche k

Etape 4 : calcul du gradient de l'erreur faite sur le neurone i dans la couche de sortie :

$$\delta_i^k = 2 \cdot f'\left(\sum_{i=1}^n (W_{ji}^k V_i^{k-1} + b_i^k)\right) \cdot (D_i^k - V_i^k)$$

Où :

f' : La dérivée de la fonction F.

D_i^k : La sortie désirée pour le neurone i dans la couche k.

Etape 5 : calcul de gradient des erreurs rétropropagées de la couche k à la couche 1.

$$\delta_i^{k-1} = f'\left(\sum_{i=1}^n (W_{ji}^k V_i^{k-1} + b_i^{k-1})\right) \cdot \sum_{i=1}^n \delta_i^k W_{ji}^k$$

Etape 6 : ajustement des poids

$$W_{ji}^k(t+1) = W_{ji}^k(t) - \eta \delta_j^{k-1}$$

Étape 7 : répétition de toutes les étapes, sauf la première, chaque fois pour un nouveau exemple, jusqu'à l'activation de test d'arrêt (convergence).

II.8.2 L'apprentissage non supervisé

C'est un apprentissage sans professeur, qui consiste à ajuster les poids à partir d'un seul ensemble d'apprentissage formé uniquement de données (les entrées). Aucun résultat désiré n'est fourni au réseau. Le réseau va de lui-même catégoriser les variables d'entrées ; une sorte d'analyse de données : on cherche à regrouper les données, selon des critères de ressemblance qui sont inconnus a priori.

Les réseaux de neurones à apprentissage non supervisé les plus utilisés sont les cartes auto-organisatrices (SOM, Self-Organization Map) [KOH 82].

II.9 Apprentissage et généralisation

La méthode de minimisation de la fonction de coût conduit à concevoir des modèles de complexités différentes et à choisir celui qui est susceptible d'avoir les meilleures propriétés de généralisation ; c'est-à-dire fournir un modèle capable de prédire la valeur de la sortie sur de nouveaux exemples.

Le meilleur modèle réalise un compromis entre la précision de l'apprentissage et la qualité de la généralisation. Si le modèle sélectionné est trop peu complexe, l'apprentissage et la généralisation sont peu précis ; si le modèle est trop complexe, l'apprentissage est satisfaisant, mais la généralisation ne l'est pas, ainsi le modèle est surajusté. Ce compromis entre la qualité de l'apprentissage et celle de la généralisation, gouverné par la complexité du modèle, est connu sous le terme de dilemme *biais-variance* [GEM 92] : Quand la complexité du modèle augmente, le biais diminue (le modèle "colle" mieux aux données) mais la variance augmente (le modèle change beaucoup lorsqu'on considère une autre base d'apprentissage). Le terme *biais* désigne la moyenne sur toutes les bases d'apprentissages possibles du carré de la différence entre les prédictions du

modèle et de la fonction de régression. Tandis que la *variance* reflète l'influence du choix de la base d'apprentissage sur le modèle.

II.9.1 Généralisation par un arrêt prématuré

L'arrêt prématuré (*early stopping*) consiste à arrêter l'apprentissage avant la convergence complète de l'algorithme. Ainsi, le modèle ne s'ajuste pas trop finement aux données d'apprentissage : le surajustement est limité. La méthode la plus classique pour choisir le moment d'arrêter l'apprentissage, consiste à suivre l'évolution de la fonction de coût sur une base de validation (une base non utilisée pour l'apprentissage), et à arrêter les itérations lorsque le coût calculé sur cette base commence à croître. (Figure II.11)

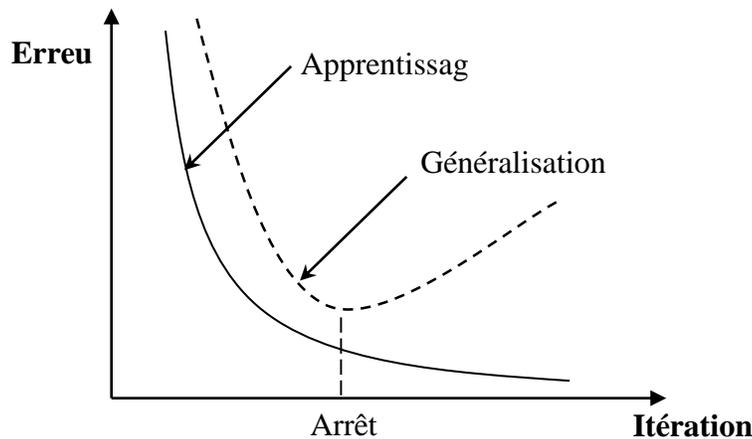


Figure II.11 : l'évolution de la fonction de coût [DU 06].

II.9.2 Généralisation par régularisation

Les méthodes de régularisation sont des méthodes de pénalisation, qui consistent à ajouter un terme à la fonction de coût usuelle afin de favoriser les fonctions régulières. La fonction à minimiser est alors de la forme : $J' = J + \alpha \cdot \Omega$ où J est, par exemple, la fonction de coût des moindres carrés. La fonction Ω la plus largement utilisée est celle qui pénalise les modèles ayant des poids élevés :

$$\Omega = \sum_i \|w_i\|^2 \quad [\text{DRE 02}].$$

L'apprentissage est réalisé en minimisant la nouvelle fonction J' . Un modèle qui a bien appris la base d'apprentissage correspond à une valeur faible de J . la méthode du *weight decay* est souvent

utilisée, car elle est simple à mettre en œuvre, et plusieurs études ont montré qu'elle conduisait à de bons résultats (voir par exemple [REE 95], [DU 06])

II.9.3 Régularisation par modération des poids (Weight-decay)

Lorsque les poids du réseau sont grands en valeur absolue, les sigmoïdes des neurones cachés sont saturées, si bien que les fonctions modélisées peuvent avoir des variations brusques. Pour obtenir des fonctions régulières, il faut travailler avec la partie linéaire des sigmoïdes, ce qui implique avoir des poids dont la valeur absolue est faible.

La méthode de régularisation du *weight decay* limite la valeur absolue des poids en utilisant

$\Omega = \frac{1}{2} \sum_{i=1}^q w_i^2$. Donc, $J' = J + \frac{\alpha}{2} \sum_{i=1}^q w_i^2$ où q est le nombre de paramètres du réseau, et α est un

hyperparamètre dont la valeur doit être déterminée par un compromis : Si α est trop grand, les poids tendent rapidement vers zéro, le modèle ne tient plus compte des données. Si α est trop petit, le terme de régularisation perd de son importance et le réseau de neurones peut donc être sur ajusté.

Le principe de la mise en œuvre de la méthode est simple : on calcule le gradient de la fonction de coût J par rétropropagation, puis on lui ajoute la contribution du terme de régularisation :

$$\nabla J' = \nabla J + \alpha w$$

Pour tenir compte du caractère différent des poids en fonction des couches, il faut considérer plusieurs hyperparamètres :

$$J' = J + \frac{\alpha_0}{2} \sum_{w_0} w_i^2 + \frac{\alpha_1}{2} \sum_{w_1} w_i^2 + \frac{\alpha_2}{2} \sum_{w_2} w_i^2$$

Où W_0 représente l'ensemble des paramètres reliant les biais aux neurones cachés, où W_1 représente l'ensemble des poids reliant les paramètres aux neurones cachés, et W_2 l'ensemble des paramètres reliés au neurone de sortie (y compris le biais du neurone de sortie). Le modèle comprend trois hyperparamètres $\alpha_1, \alpha_2, \alpha_3$ qui doivent être déterminés. Une démarche heuristique, consiste à effectuer plusieurs apprentissages avec des valeurs différentes des paramètres, à tester les modèles obtenus sur un ensemble de données de validation, et à choisir le meilleur.

II.10 Sélection de modèle

Comme indiqué dans la section précédente, la sélection de modèles est une étape cruciale dans la conception d'un modèle par apprentissage. Nous allons présenter les méthodes les plus fréquemment utilisées :

II.10.1 La validation croisée (cross-validation)

La validation croisée est une méthode d'estimation de l'erreur de généralisation d'un modèle à partir de données qui ne sont pas utilisées en apprentissage [JAN 88]. On divise l'ensemble des données disponible en D sous ensembles disjoints. On effectue ensuite les étapes suivantes :

1. itération i à réaliser D fois : construire une base d'apprentissage par l'union de $D-1$ sous-ensemble ; effectuer plusieurs apprentissages, avec des initialisations différentes des poids, avec cette base d'apprentissage ; pour chacun des modèles obtenus, calculer l'erreur quadratique moyenne sur l'ensemble de validation (EQMV) par le sous ensemble des N_v

exemples restants $E_v = \sqrt{\frac{1}{N} \sum_{k=1}^{N_v} (y_k^p - g(x_k, w))^2}$, retenir la plus petite de ces erreurs E_{vi} .

2. calculer le score de validation croisée à partir des D quantités E_{vi} au cours de ces D

itérations $\sqrt{\frac{1}{D} \sum_{i=1}^D (E_{vi})^2}$; ce score constitue une estimation de l'erreur de généralisation de

la famille de modèle considéré.

Ainsi, si l'on choisit $D=5$ (ce qui est une valeur typique appelée en anglais *5-fold cross-validation*) ; on réalise 5 partitions différentes de la base de données, dans chacune de ces partitions 80% des données sont dans l'ensemble d'apprentissage et 20 % dans l'ensemble de validation (figure II.10). On sélectionne le modèle ayant le plus faible score de validation.

Cette technique a l'avantage d'être moins sensible au choix des bases de validation, et permet d'utiliser toutes les données en apprentissage. Elle nécessite cependant d'effectuer d'apprentissages

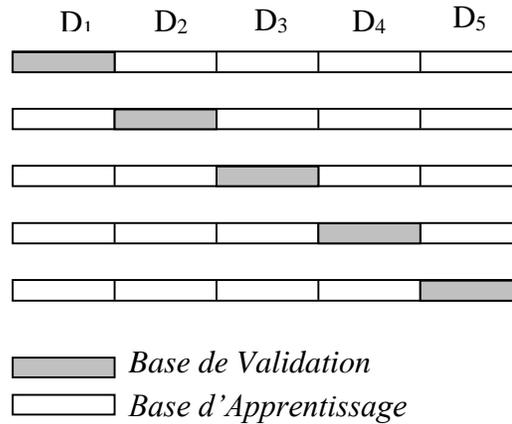


Figure II.12 : Principe de la validation croisée

II.10.2 La méthode Leave-one-out

L'estimation de l'erreur de généralisation par leave-one-out est un cas particulier de la validation croisée, pour laquelle $D=N$: à l'itération k de l'ensemble d'apprentissage, on effectue des apprentissages (avec des initialisations différentes des poids) avec les $N-1$ élément de la base d'apprentissage ; pour chacun des modèles obtenus, on calcule l'erreur de prédiction commise sur l'observation k lorsque celle-ci est extraite de l'ensemble d'apprentissage, et l'on retient la plus petite de ces erreurs, notée $R_k^{(-k)}$. Le score de leave-one-out est alors défini par

$$E_l = \sqrt{\frac{1}{N} \sum_{k=1}^N (R_k^{(-k)})^2} .$$

On utilise ce score, comme dans le cas de la validation croisée, en

augmentant progressivement la complexité des modèles.

II.11 Conclusion

Nous avons présenté dans ce chapitre les réseaux de neurones MLP comme outils pour la régression non-linéaire. Nous avons discuté de l'apprentissage de ces modèles paramétriques et du problème de la généralisation liée au principe de minimisation du risque empirique. Nous avons présenté notamment les méthodes classiques pour augmenter la généralisation d'un modèle d'apprentissage.

Ces réseaux sont maintenant considérés comme des outils mathématiques, indépendamment de toute référence à la biologie ; permettant de résoudre, avec une grande précision, les problèmes de modélisation ou d'identification linéaires et non-linéaires.

Chapitre 3 :

LA Réalisation

III.1. Introduction :

Après avoir achevé le chapitre2, nous allons entamer dans ce chapitre la partie réalisation et implémentation dans laquelle on s'assure que le système est prêt pour être exploité par les utilisateurs. A la fin de ce chapitre, notre travail nous permettra d'atteindre nos objectifs cibles.

Ce Project a pour but de faire l'apprentissage et la prédiction des propriétés de PEBD en utilisant les réseaux de neurone.

Avant de présenter l'application nous allons présenter les outils nécessaires pour ce projet.

III.2. Les choix techniques :

Afin de mener à bien notre projet nous avons utilisé comme outils de développement l'outil Matlab dans le but d'avoir les poids ciblés ainsi que le IDE SharpDevelop pour développer notre application toute on utilisant le langage c#.

III.2.1. Matlab :

MATLAB est un langage de programmation de quatrième génération et l'environnement de l'analyse numérique.

Les utilisations de MATLAB comprennent des calculs matriciels, le développement et l'exécution des algorithmes, la création d'interfaces utilisateur (UI) et la visualisation des données.



Figure III.1 : logo MATLAB

III.2.2. C# :

C # (prononcé "C-sharp") est un langage de programmation orienté objet de Microsoft qui vise à combiner la puissance de calcul de C ++ avec la facilité de programmation de Visual Basic. C # est basé sur C ++ et contient des fonctions similaires à celles de Java.



Figure III.2 : logo C#

III.2.3. SharpDevelop:

SharpDevelop (autrement écrit #develop) est un environnement de développement intégré Open Source, et permet le développement d'application en Visual Basic .NET et en C#. Il fait partie, avec MonoDevelop des alternatives à Visual Studio.Net de Microsoft.

SharpDevelop permet de développer des applications en utilisant le framework .NET de votre choix (2.0, 3.0, 3.5 ou 4.0).



Figure III.3: logo SharpDevelop

III.3. Les interfaces de l'application :

Notre application est capable d'effectuer un apprentissage et une prédiction avec précision et la possibilité de tracer des graphes.

III.3.1. Création d'un réseau du neurone :

Notre application permet à l'utilisateur de créer un nouveau réseau de neurone ou bien ouvrir un réseau déjà existant. Pour créer un nouveau réseau on doit tout d'abord appuyer sur le bouton File puis Nouveau. Une fenêtre s'affiche au milieu on fait entrer le nombre de couche et la fonction d'activation comme le montre la figure (**Figure III.4,5**), Ensuite on doit entrer le nombre de neurone pour chaque couche.

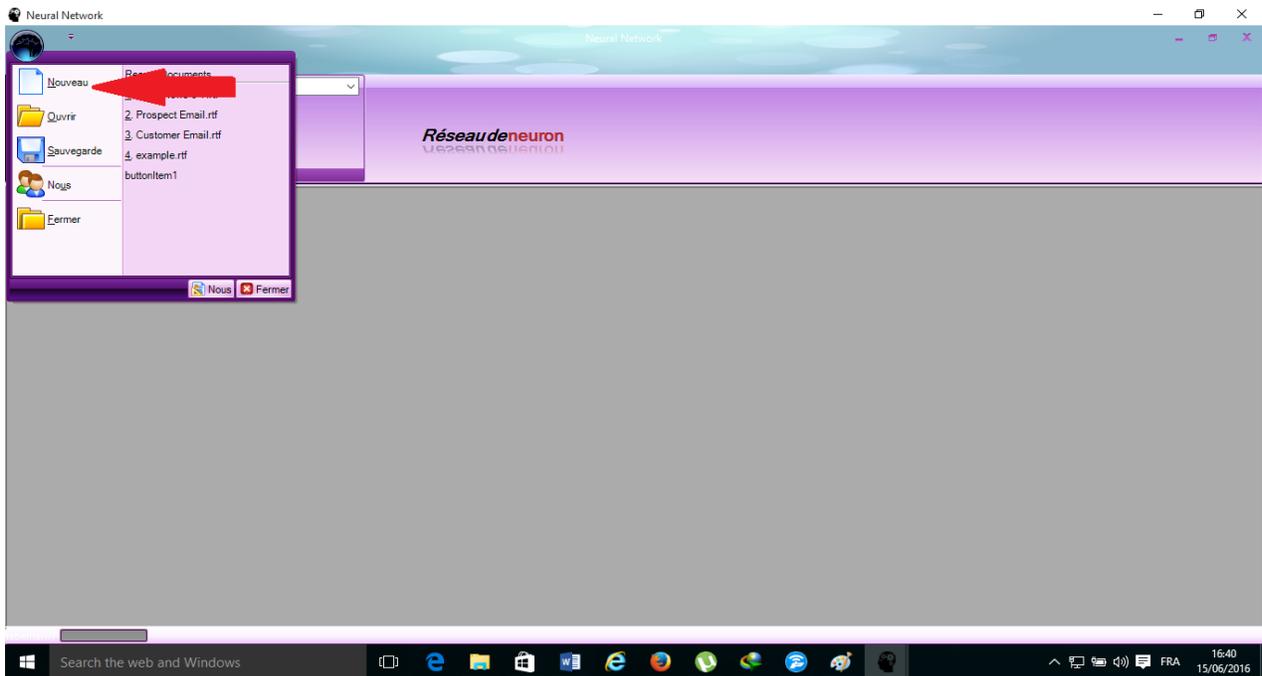


Figure III.4 : création d'un nouveau réseau de neurone

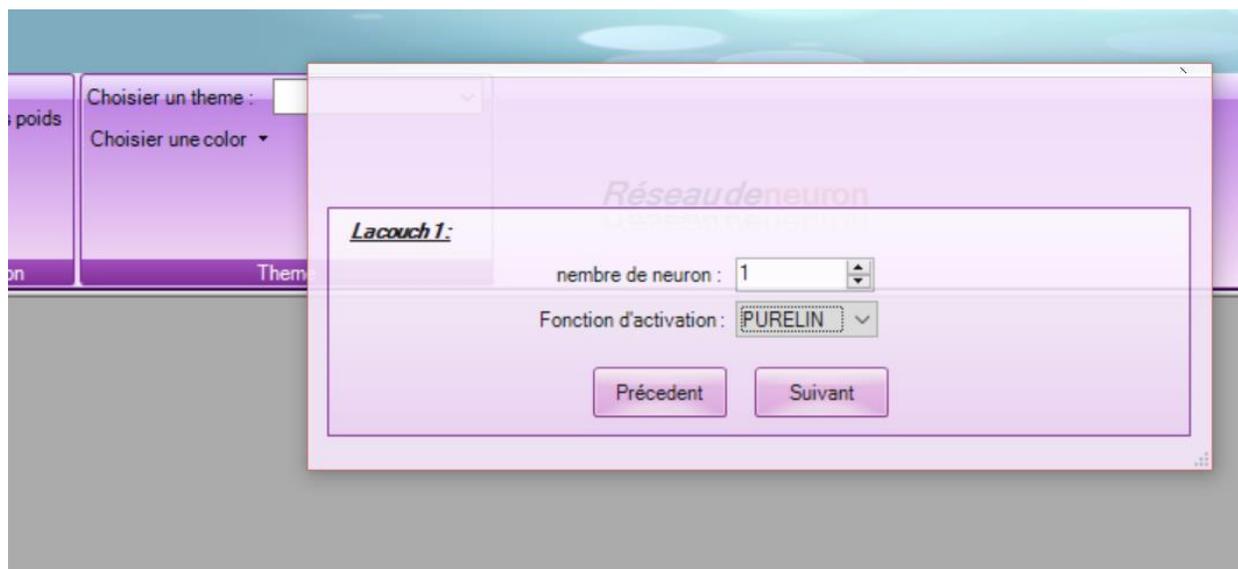


Figure III.5 : choix du nombre de neurone et de la fonction d'activation

L'utilisateur peut parcourir les fichiers et importer les données comme le montre la Figure III.6

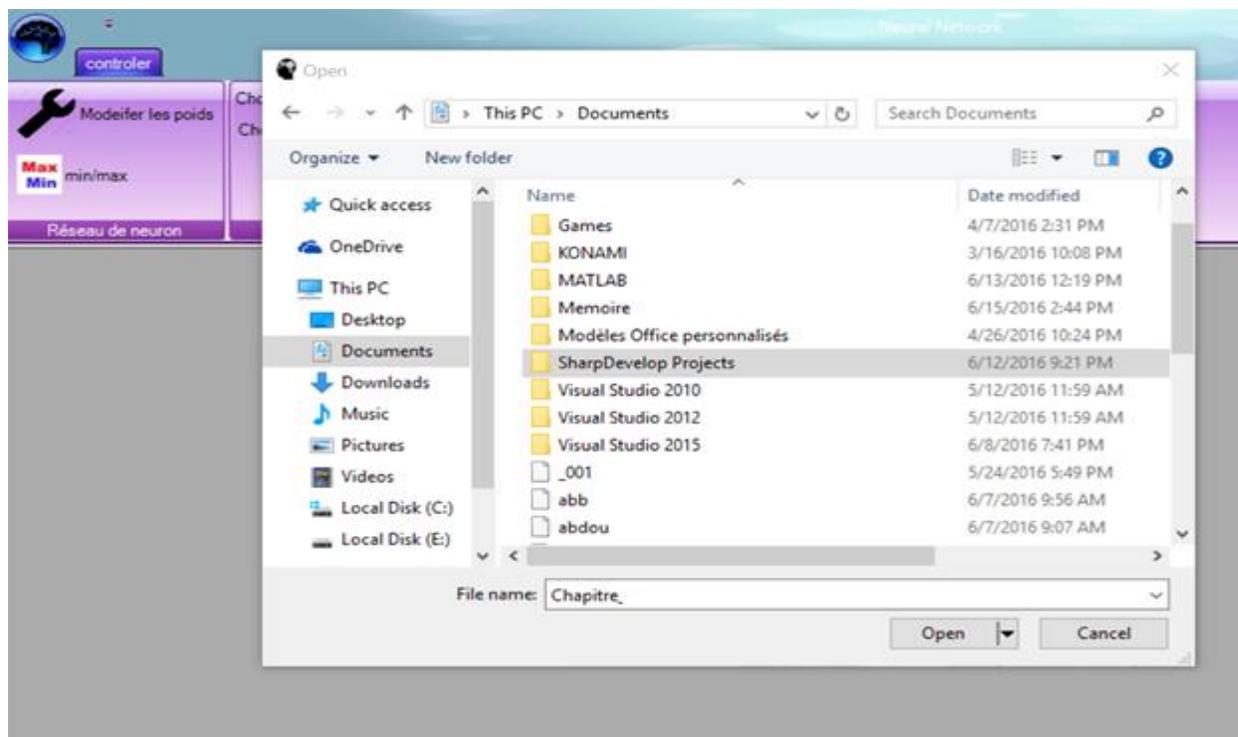


Figure III.6 : Ouvrir un nouveau réseau de neurone

III.3.2. Sauvegarde un réseau du neurone :

L'utilisateur peut sauvegarder le réseau toute en indiquant l'endroit et le nom de fichier.

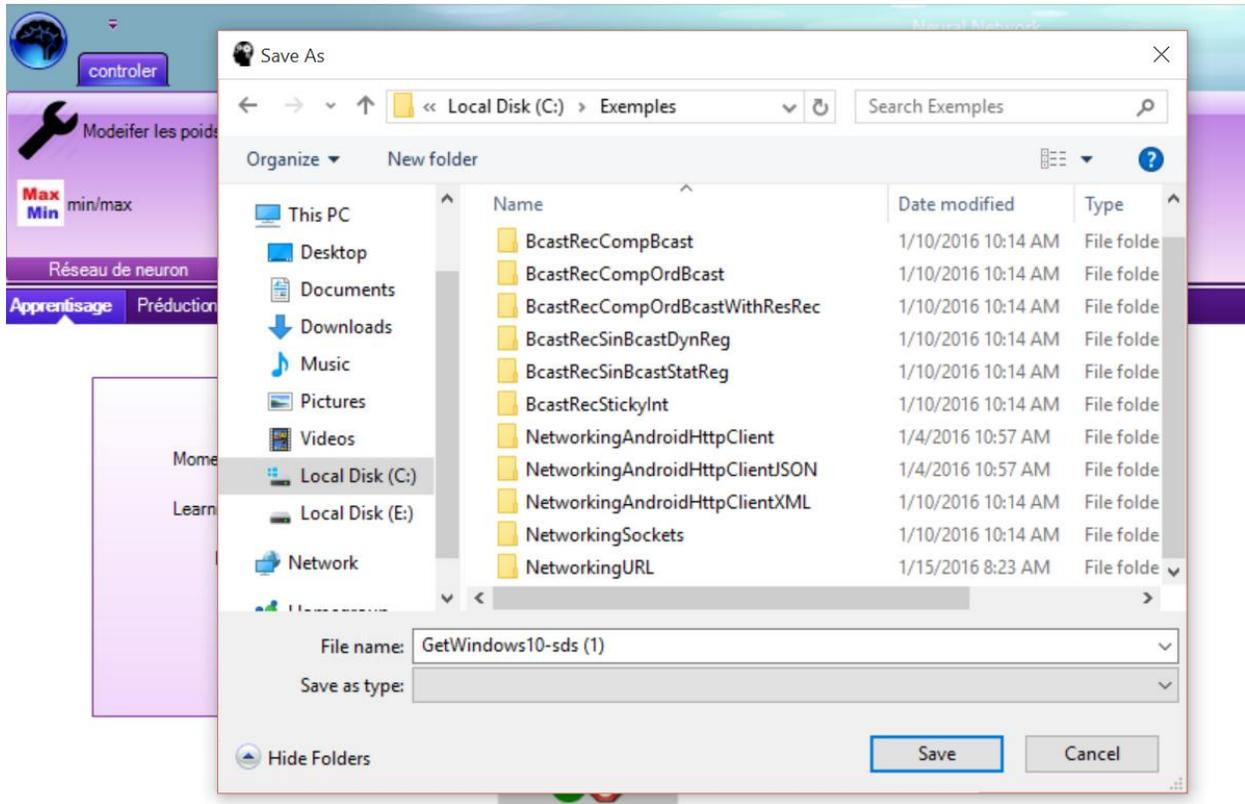


Figure III.7: Sauvegarde un nouveau réseau de neurone

III.3.3. Modifier les paramètres de réseau de neurone :

L'utilisateur à la possibilité de modifier les paramètres de réseau à savoir :

III.3.3.1. Modifier les poids :

Notre application permet de préciser les poids ainsi que les biais. On note que la dernière colonne concerne les biais.

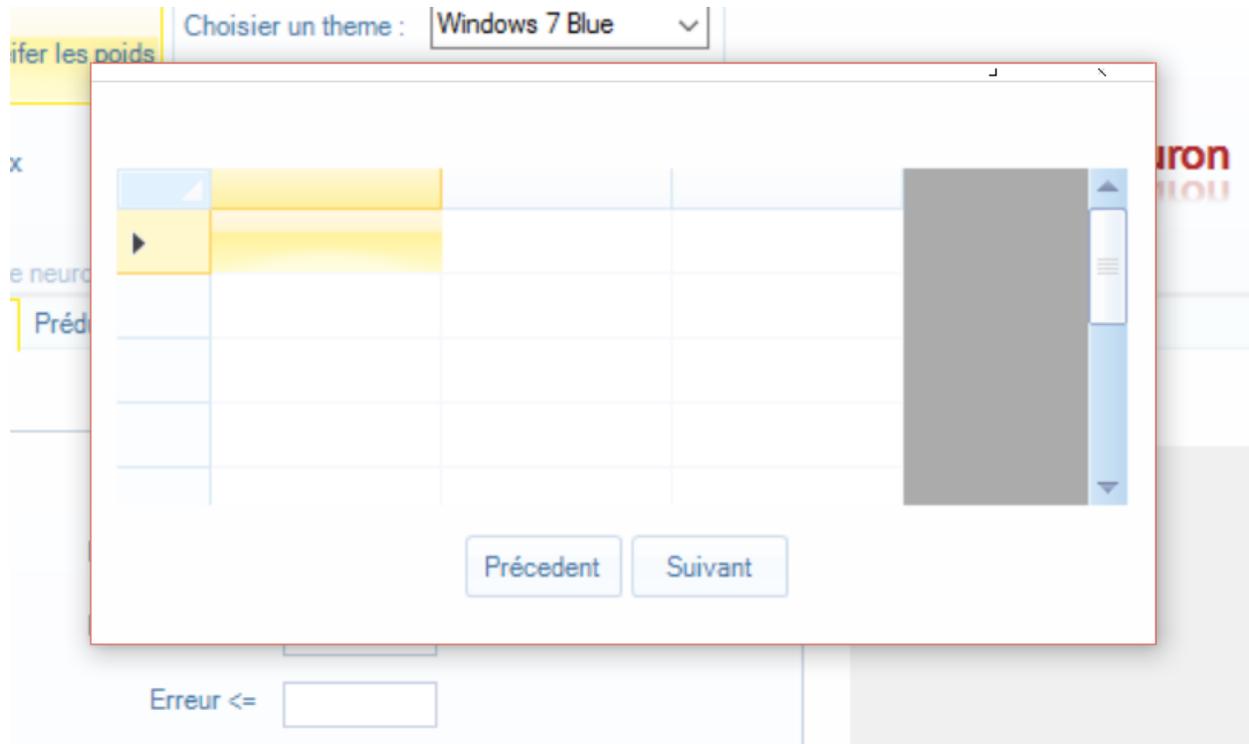


Figure III.8: fenêtre de modification des poids et biais

III.3.3.2. Modifier les valeurs min/max des entrées et de la sortie :

L'utilisateur peut déterminer les valeurs min/max des entrées et de la sortie.



Figure III.9 : fenêtre de modification des min/max

III.3.4. L'apprentissage :

Après avoir créé un réseau de neurone l'utilisateur peut effectuer un apprentissage en chargeant un fichier Excel qui contient l'entrée associée à leur sortie.

Il peut aussi spécifier l'erreur à laquelle l'algorithme doit s'arrêter et le taux d'apprentissage

Pour spécifier à quelle valeur de l'apprentissage doit se terminer. Comme le montre la figure (Figure III.9).

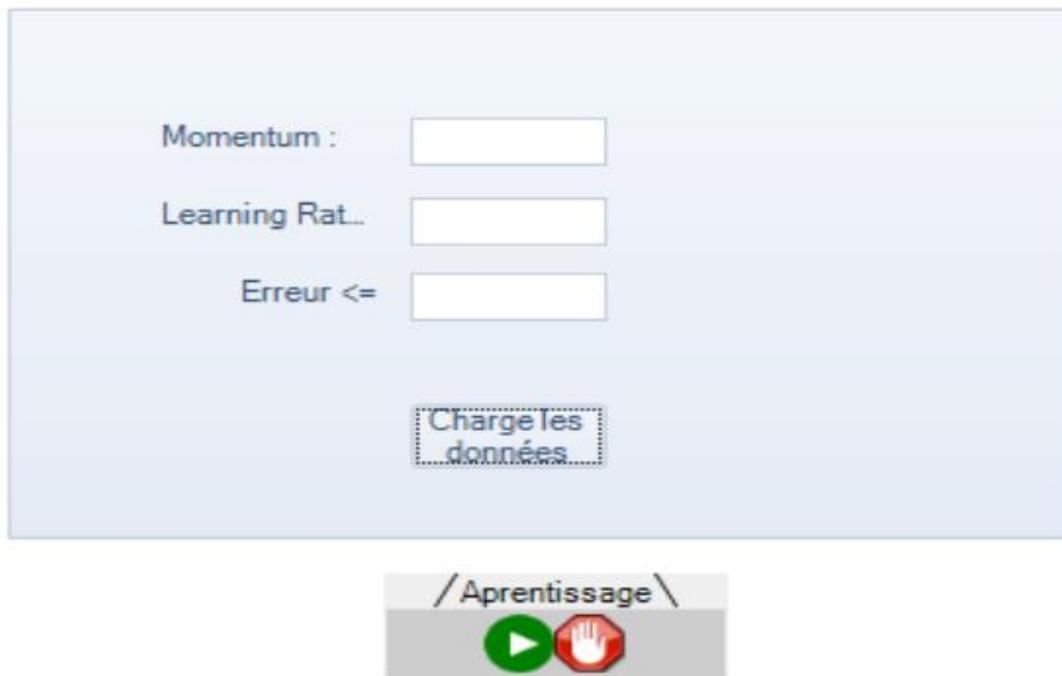


Figure III.10 : fenêtre d'apprentissage

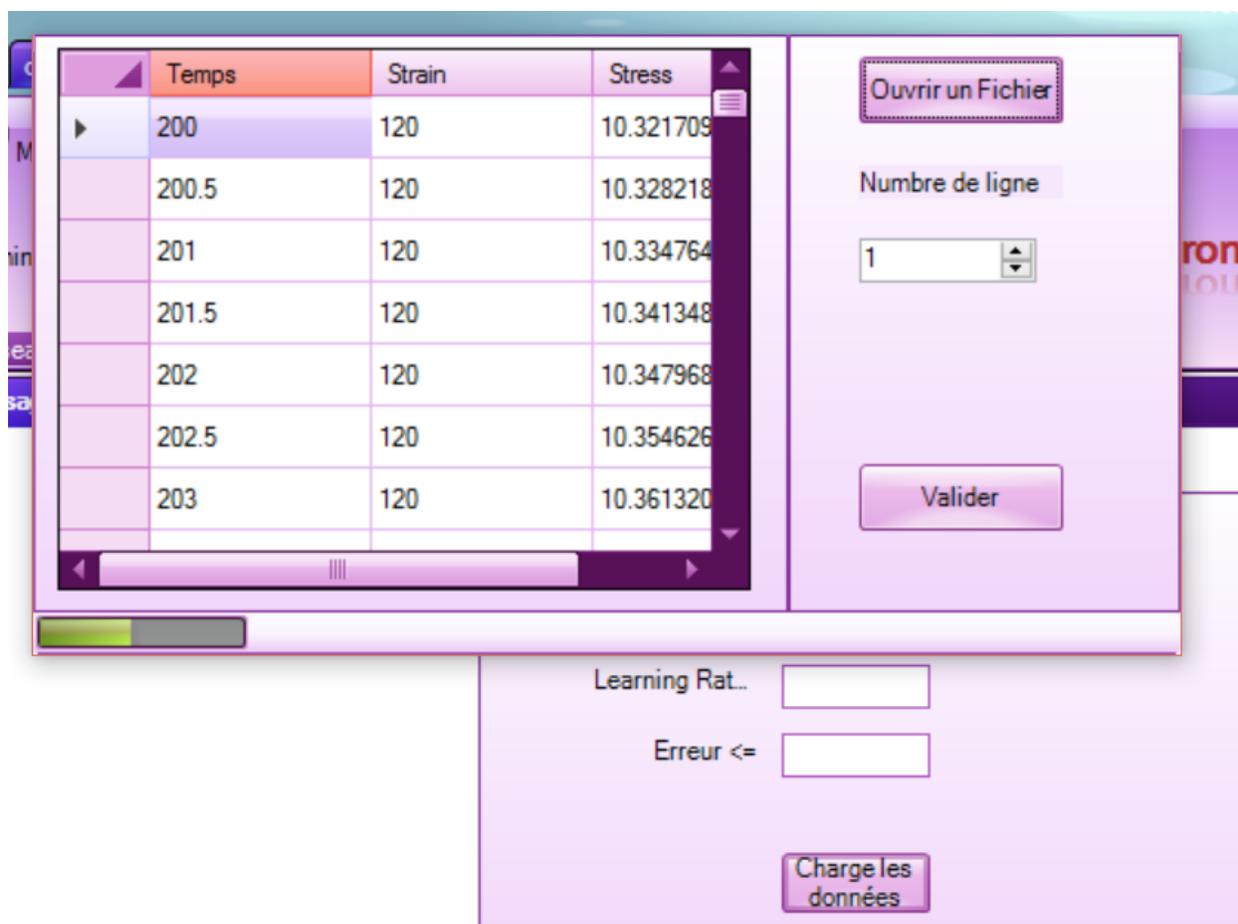


Figure III.11 : fenêtre de chargement d'un fichier Excel

III.3.5. La prédiction :

La prédiction peut être effectuée soit sur une paire de données ou bien sur un ensemble de données.

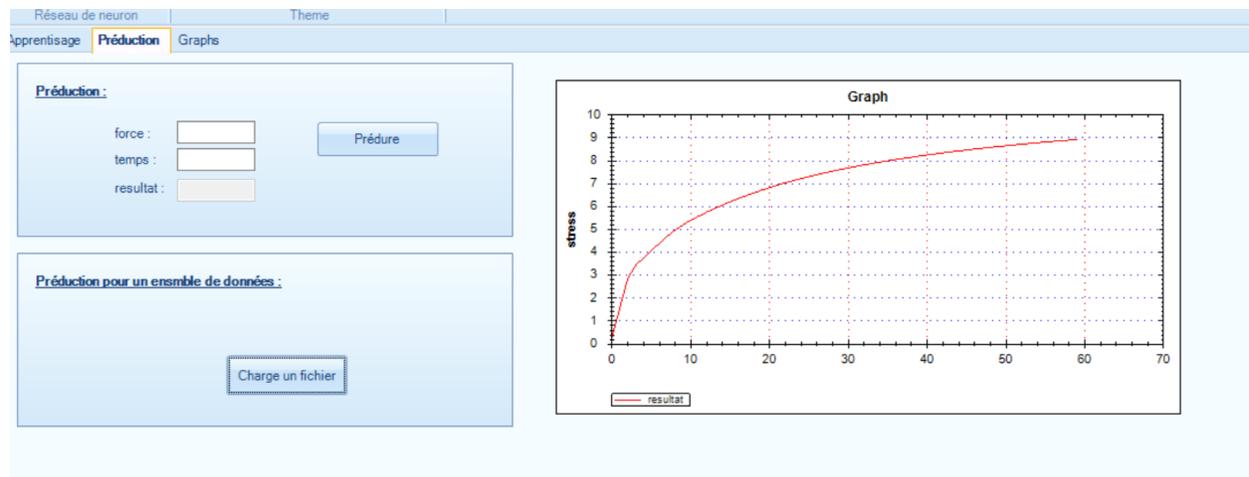


Figure III.12 : fenêtre de prédiction

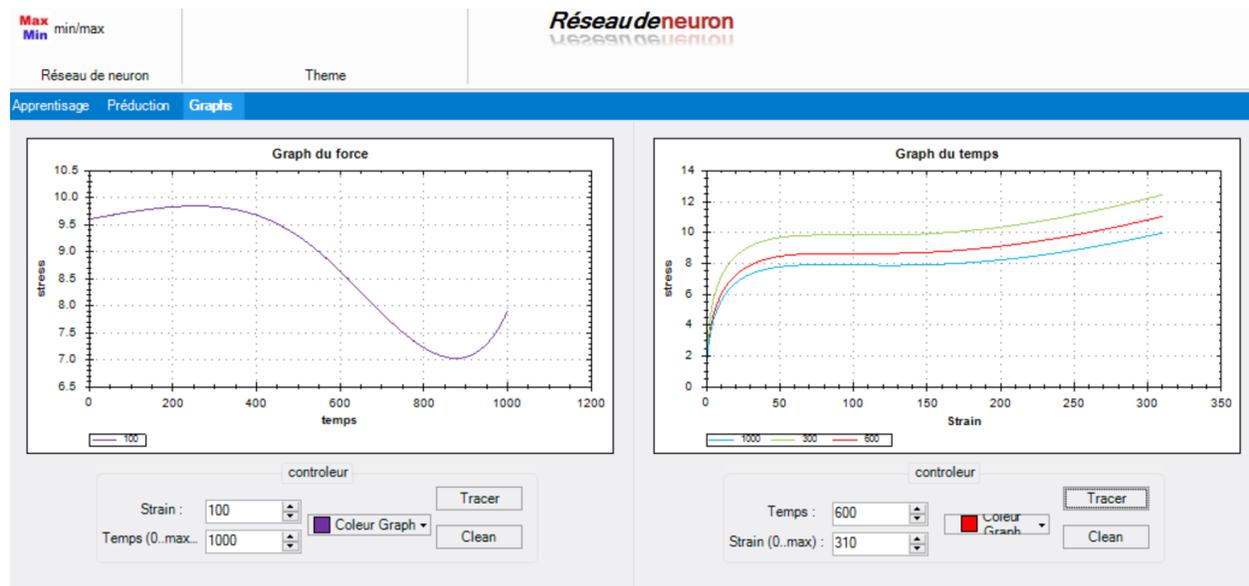


Figure III.13 : fenêtre de prédiction

Le premier graph démontre les stress en fonction de temps et le deuxième graph le stress en fonction de strain.

III.4. Les résultats :

III.4.1. Les tests :

NB: Le graph en bleu représente les donnée expérimentale et le graph en rouge représente les résultats simulée par le réseau.

Pour temps = 120 :

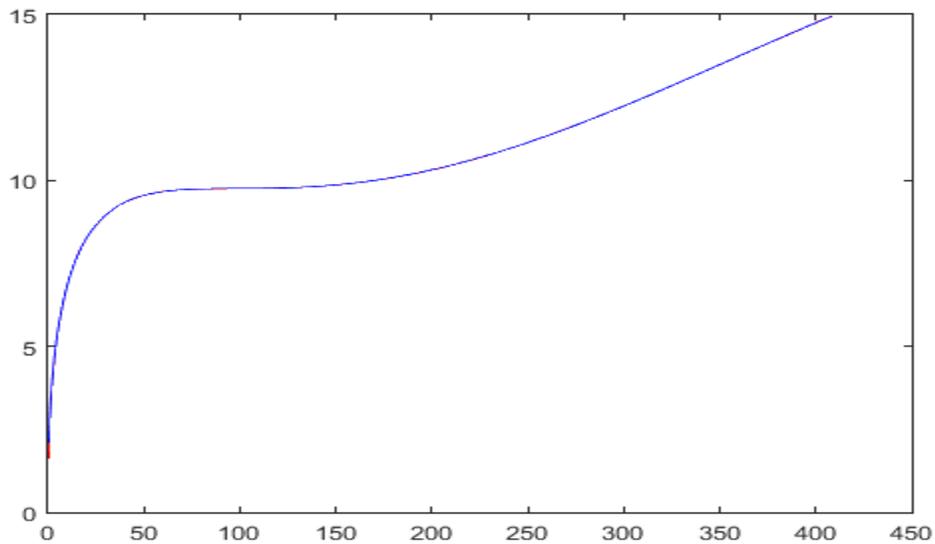


Figure III.14 : graphe de test pour temps =120

Pour temps = 1010 :

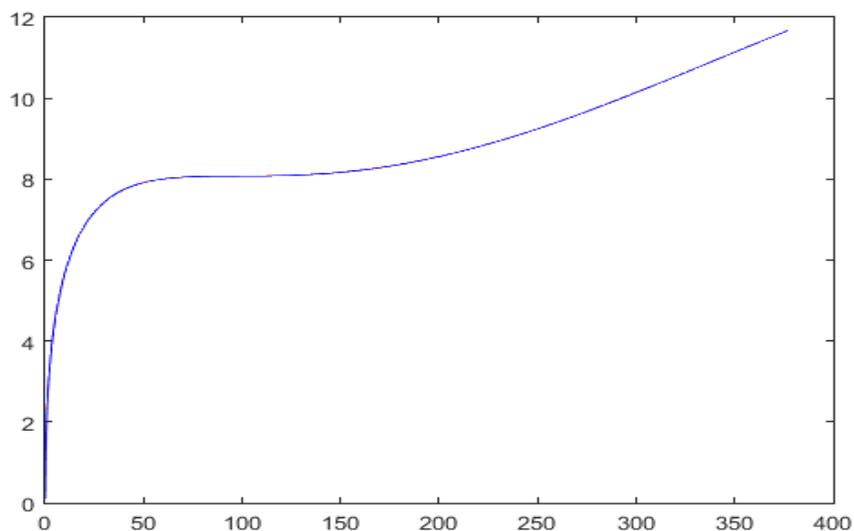


Figure III.15 : graphe de test pour temps =1010

Pour strain = 10 :

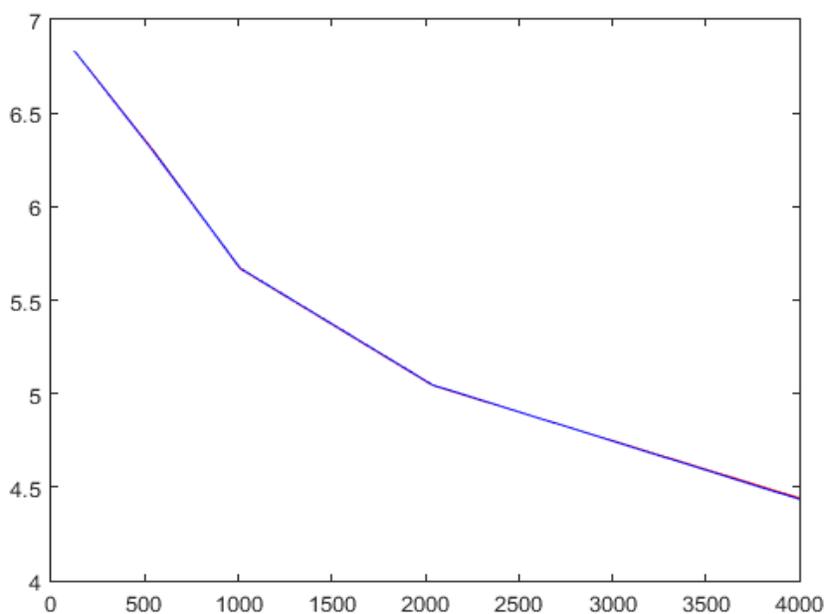


Figure III.16 : graphe de test pour strain =1010

Pour strain = 200 :

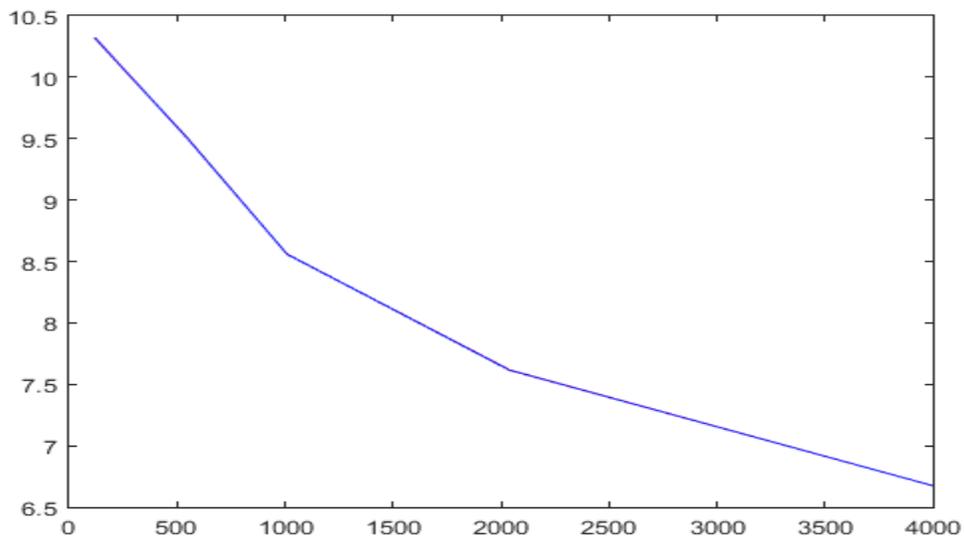


Figure III.17 : graphe de test pour strain =200

III.4.1. Prediction:

Pour temps = 300 :

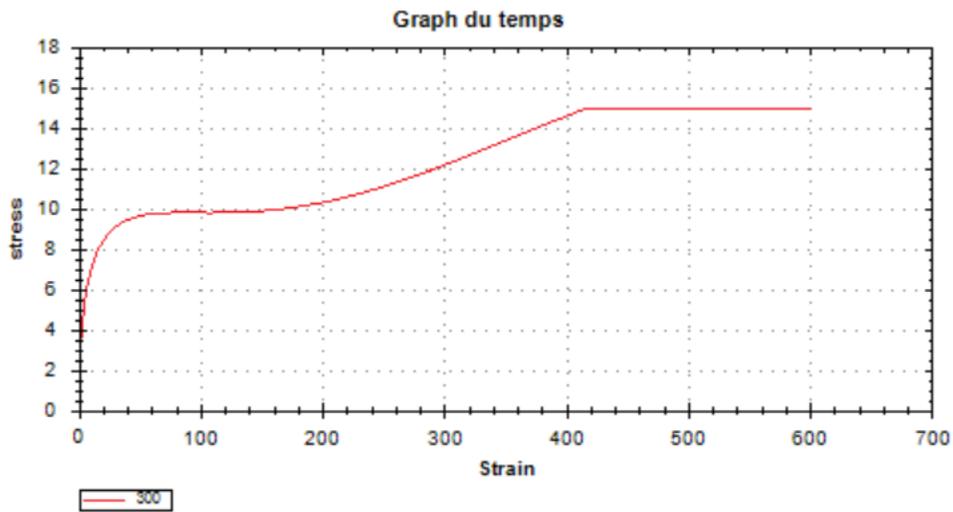


Figure III.18 : graphe de prédiction pour temps =300

Pour temps = 60 :

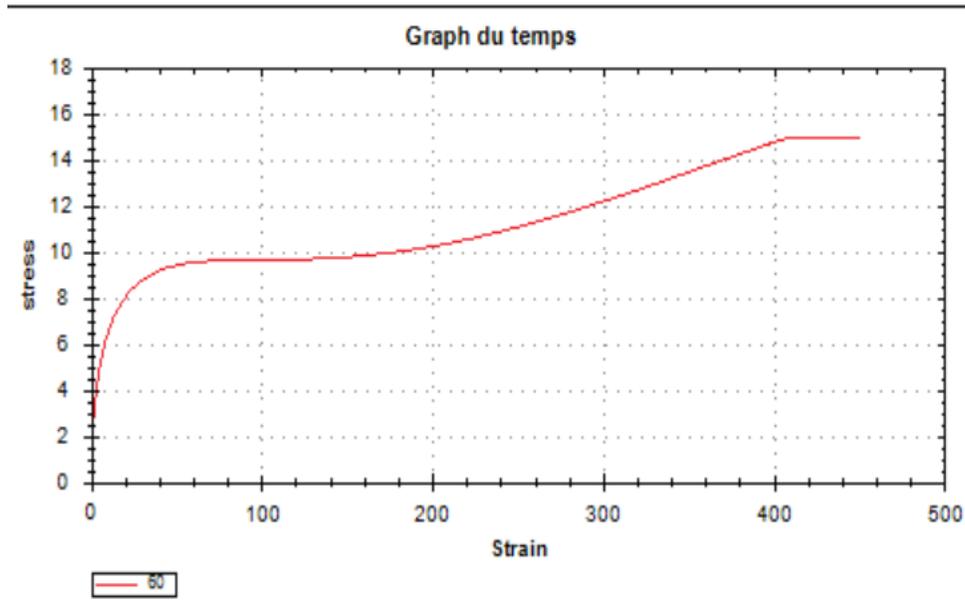


Figure III.19 : graphe de prédiction pour temps =60

Pour strain = 500 :

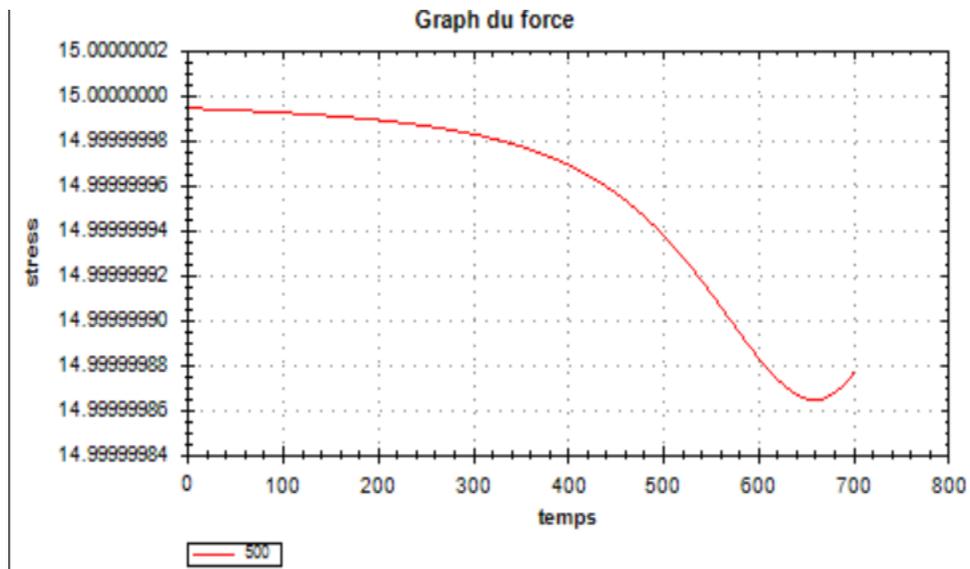


Figure III.20 : graphe de prédiction pour strain =500

Pour strain =700 :

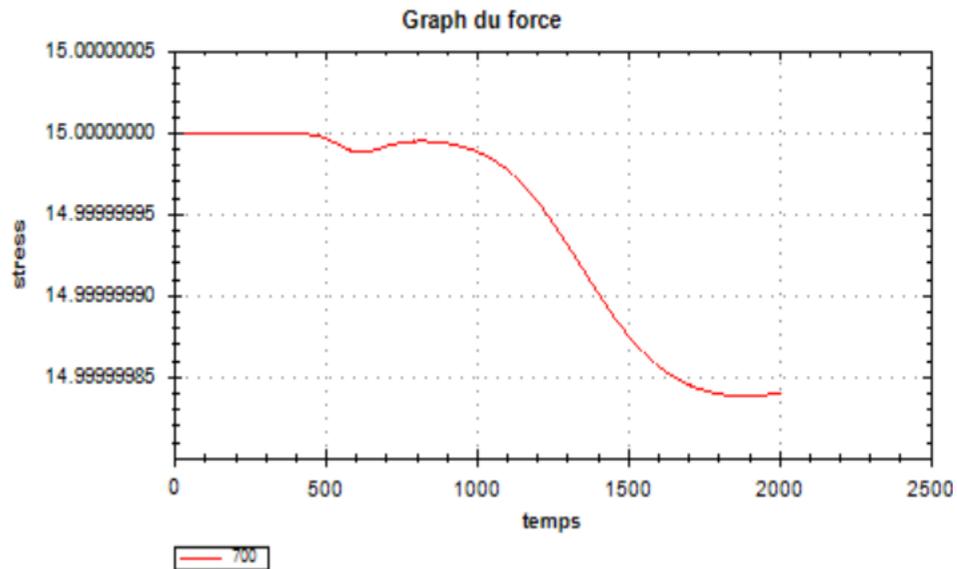


Figure III.21 : graphe de prédiction pour strain =700

III.5. Conclusion :

Dans ce chapitre nous avons, en premier lieu, présenté les différents outils et langages que nous avons utilisés pour implémenter notre application. Et présenter quelques interfaces de notre application en expliquant clairement comment manipuler l'application ainsi que les résultats des tests et de la prédiction.

Conclusion générale :

Dans de nombreux secteurs d'activités, le vieillissement doit être maîtrisé sinon il constitue un obstacle important à l'utilisation des matériaux polymères. Il n'y a pas un vieillissement des polymères mais des vieillissements, ceux-ci peuvent avoir une origine physique, chimique et les différents mécanismes sont souvent liés ou induits les uns par les autres. La conception d'un produit en matériau polymère doit prendre en compte les effets du vieillissement de la matière sur les fonctions premières.

Ce travail porte en premier temps sur la modélisation physique et mécanique du phénomène de vieillissement d'un isolant organique (PEBD).

Les polymères ont envahi ce domaine par excellence. Le succès et l'utilisation massive et intense de ces polymères s'accompagnent malheureusement de risques encourus suite à leur dégradation.

Le premier but de ce travail est la compréhension du processus de vieillissement des polymères qui, de par leur utilisation, sont soumis à des contraintes thermiques et mécaniques. Dans ce but, nous nous sommes d'abord intéressés aux propriétés structurales des polymères qui sont pour la majorité semi-cristallins. En effet, ces matériaux présentent une structure spécifique qui dicte leurs propriétés physiques, mécaniques. La prise en compte de cette microstructure est donc nécessaire pour décrire leur comportement mécanique.

En raison de propriétés mathématiques qui sont maintenant bien comprises, et grâce au développement de méthodologies rigoureuses pour la conception de modèles, les réseaux de neurones sont devenus des outils de modélisation puissants dont les domaines d'application sont multiples. Ils permettent de réaliser, de manière simple et efficace, des modèles précis, statiques ou dynamiques, parcimonieux en nombre de paramètres. Contrairement à une idée répandue, ils ne sont pas nécessairement des «! boîtes noires !» sans signification physique ! la méthodologie de modélisation semi-physique permet de tirer la meilleure partie possible des données expérimentales disponibles et des connaissances avérées, sous formes d'équations algébriques ou différentielles, qui peuvent résulter d'une étude physique du processus que l'on cherche à modéliser.

Dans ce présent travail, notre deuxième objectif est l'utilisation de nouvelles stratégies –à base de réseaux de neurones artificiels– de commande afin d'améliorer le control et la prédiction du déclin de potentiel de surface d'un PEBD. Ces derniers sont des techniques puissantes de traitement non-linéaire de données et qui ont fait leurs preuves dans de nombreux domaines d'application à caractère industriel. Ceux-ci (les RNAs) disposent de règles bien précises pour fixer le nombre de neurones et de couches cachées.

On a eu en premier lieu fait une comparaison entre les deux algorithmes de Levenberg-Marquardt et Scaled conjugate gradient, aussi bien entre différentes architectures en changeant le nombre de couche cachée, afin de fixer l'architecture optimale du réseau.

Il apparaît que le réseau RNA choisi (un réseau à une seule couche cachée entraînant par l'algorithme de Levenberg-Marquardt) a présenté de bons résultats, avec des erreurs allant jusqu'à 0.0030 en généralisation.

Ce travail est une première étape dans le développement d'un modèle neuronal complet capable de prédire le déclin de potentiel de surface d'un PEBD, nous avons défini seulement le nombre de couche cachée et le nombre de neurones par couche. Après un ensemble d'essais (4*1) nous avons obtenu un seul modèle neuronal présentant les meilleurs résultats. Ensuite, nous avons réalisé une analyse comparative de ces différents modèles par un test comprenant des exemples n'appartiens pas de la banque de données utilisée pour les trois phases d'apprentissage, de validation et de généralisation.

Nous souhaitons compléter l'étude par le développement d'un autre logiciel sur les algorithmes génétiques. Comme nous espérons faire une étude comparative sur la simulation des modèles avec les réseaux de neurones et les algorithmes génétiques.

Enfin, nous souhaitons que ce modeste travail serve comme référence bibliographique aux chercheurs qui s'intéressent au domaine de la commande neuronale.

Bibliographie :

- [1] Arbolil.M.Plasticulture n°119(2000) p15-25 les plastiques pour la couverture des serres et abris.
- [2] polymer degradation and stability ,68(2000)465-469 comparison of formation of oxidation products during photo –oxidation of low density polyethylene under different natural and accelerated weathering conditions.
- [3] G.et Bonora M.Plasticulture n°119(2000) p83=101 Plasticulture pour serres avec stabilisants HALS Améliorés.
- [4] J.Bost , Matières Plastiques ,I-Chimie Application ,2eme Edition ,Techniques et Documentations (1985).
- [5] C.Bord, Guide des Matières plastiques en Mécaniques ; Tome 1 et Tome 2 .CETIM(1976).
- [06] Patricia nicaud. Projet troisième rob'ok
- [07] REACH for Polymers ,meilleures méthodes d'essai et techniques disponibles 2011, smithers Rapra
- [08] M.Fontanille, Y.Gnanou et J.M.Lehn; « Chimie et Physico-chimie des Polymères » DUNOD Belgique 1^{re} édition; 2005.
- [09] J.DorlotT, J. Massounave et al, "des matériaux", edition de l'ecole polytechnique des matériaux, Juillet, (1995).
- [10] S.Fuzessery, Techniques de L'Ingénieur (1988) . matériaux, Juillet, (1995).
- [11] J. Martin et al, " traité des matériaux ", Tome 14, (2001).
- [12] Mahrez-BERRANEM « Etude du vieillissement sur les propriétés physiques et mécaniques du polyéthylène a basse densité (PEBD) » **Année 2010**

- [13] Pan J., Shaw W.J.D., Properties of a mechanically processed polymeric material, *Journal of Applied Polymer Science*, 52 (4), 507-514 (1994)
- [14] Pan J., Shaw W.J.D., Effects of processing parameters on material properties of mechanically processed polyamide, *Journal of Applied Polymer Science*, 56 (5), 557-566 (1995)
- [15] Ludovic Douminge. Etude du comportement du polyéthylène haute densité sous irradiation ultraviolette ou sollicitation mécanique par spectroscopie de fluorescence. Engineering Sciences. Université de La Rochelle, 2010. French.
- [16] Yakimets-Pilot, I., Elaboration d'un modèle viscoélastique et son application au comportement d'un polypropylène sous sollicitations mécaniques complexes et physico-chimiques, 2004.
- [17] Dupend-Brusselle, N., Comportement viscoélastoplastique d'un polymère semi-cristalin, le polypropylène, avant la striction, 2000.
- [18] R. Gattin, C. Poulet, A. Copinet, Y. Couturier. Comparison of mineralization of starch in liquid, inert solid and compost media according to ASTM and CEN norms for the composting of packaging materials. *Biotechn. Letters*. Vol 22 (2000) 1471-1475.
- [19] I. Jakubowicz. Evaluation of degradability of biodegradable polyethylene (PE). *Polym. Degrade. Stab*. Vol 80 (2003) 39-43.
- [20] K-L. G. Ho, A. L. Pometto. Temperature effects on soil mineralization of polylactic acid plastic in laboratory respirometers. *J. Environ. Polym. Degrade*. Vol 7 No 2 (1999) 101-107.
- [21] H. Nishide, K. Toyota, M. Kimura. Effects of soil temperature and anaerobiosis on degradation of biodegradable plastics in soil and their degrading microorganisms. *Soil. Sci. Plant. Nutr*. Vol 45 No 4 (1999) 963-972.
- [22] Grossmann P.R. Connaissez votre ennemi : le temps et comment le reproduire en laboratoire. *Galvano-organo-traitements de surface*, 607, (1990) 569
- [23] S. Li, S. Mc. Carthy. Further investigations on the hydrolytic degradation of poly (DL-lactide). *Biomaterials*. Vol 20 (1999) 35-44.

- [ACK 85] D. H. Ackley, G. E. Hinton, T. J. Sejnowski, “A learning algorithm for Boltzmann machines”, *Cognitive Science*, Vol. 9, No. 1, pp. 147–169, 1985.
- [AID 11a] L.AID, M.LOUDINI, W.K.HIDOUCI, “An Admission Control Mechanism for Web Servers using Neural Network ”. *International Journal of Computer Applications*, Foundation of Computer Science, New York, USA. Volume 15–No.5, pp. 14-19, February 2011
- [ALE 10] Alexa the web information company, <http://www.Alexa.com> , juillet 2010.
- [AND 03] M. Andersson, M. Kihl and A. Robertsson, “Modeling and Design of Admission Control Mechanisms for Web Servers using Non-linear Control Theory”, In *Proceedings of ITCOM’03*, Orlando, USA, Sept. 2003.
- [AND 04] M. Andersson, “Overload Control and Performance Evaluation of Web Servers”, PhD thesis, Department of Electrical and Information Technology, Lund Institute of Technology, 2004.
- [AND 05a] M. Andersson, “Introduction to Web Server Modeling and Control Research”, Technical Report, Lund Institute of Technology, 2005.
- [AND 05b] M. Andersson, A. Bengtsson, M. Höst, C. Nyberg, “Web Server Traffic in Crisis Conditions”, In *Proceedings of the Swedish National Computer Networking Workshop, SNCNW*, 2005.
- [APA 10] “Apache web server”, <http://www.apache.org>, consulté le 13/03/2010.
- [AUT 10] <http://www.xenoclast.org/autobench>

- [BAT 88] D. M. Bates, “Nonlinear Regression Analysis and Its Applications”, Edition John Wiley & Sons, 1988.
- [BER 96] T. Berners-Lee, R. Fielding and H. Frystyk, “Hypertext Transfer Protocol – HTTP/1.0”, RFC 1945, Network Working Group, 1996.
- [BRA 98] P. Barford, M. Corvella, “Generating Representative Web Workloads for Network and Server Performance Evaluation”, Proceedings of the ACM SIGMETRICS’98, 1998, pp. 151-160.
- [CAO 03] J. Cao, M. Andersson, C. Nyberg, M. Kihl, “Web Server Performance Modeling Using an M/G/1/K*PS Queue”, (Extended version), In Proceedings of the 10th International Conference on Telecommunications, Feb. 2003, Papeete, Tahiti.
- [COM 10] Complete Survey, <http://www.complete.com> , juillet 2010.
- [CHE 90] D. L. Chester, “Why two hidden layers are better than one”, In Proc. of the Int. Joint Conf Neural Net., Washington DC, 265–268, 1990.
- [CHE 01] X. Chen, P. Mohapatra, H. Chen, “An Admission Control Scheme for Predictable Server Response Time for Web Accesses”, In Proceedings of the International World Wide Web Conference (10), Hong-kong, 2001
- [COM 98] D. Comer, “TCP/IP Architecture, Protocoles, Applications”, 3^{ème} édition, InterEditions, 1998.
- [CYB 89] G. Cybenko, “Approximation by superposition of a sigmoid function”, Math. of Contr, Signals & Syst., Vol. 2, pp. 303–314, 1989.

- [DIA 02] Y. Diao, N. Gandhi, J. L. Hellerstein, S. Parekh, and D.M. Tilbury, “Using MIMO feedback control to enforce policies for interrelated metrics with application to the Apache Web server”, Network Operations and Management Symposium, 2002.
- [DRE 02] G. Dreyfus, J.M. Martinez., M. Samuelides, M.B. Gordon, F. Badran, S. Thiria et L. Hérault, “Réseaux de neurones, méthodologie et applications”, Eyrolles, (2002).
- [DU 06] K.-L. Du and M. N. S. Swamy, “Neural Networks in a Softcomputing Framework”, Springer-Verlag London Limited, 2006.
- [DUC 99] W. Duch, N. Jankowski N (1999) “Survey of neural transfer functions”. Neural Comput Surveys 2:163–212
- [ELN 04] S. Elnikety, E. Nahum, J. Tracey, W. Zwaenepoel: “A Method for Transparent Admission Control and Request Scheduling in E-Commerce Web Sites”. In Proceedings of the 13th International Conference on World Wide Web, 2004.
- [FIE 99] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach and T. Berners-Lee, Hypertext Transfer Protocol – HTTP/1.1, RFC 2616. Internet Engineering Task Force (1999).
- [FLA 10] “Flash web server,” [http : //www.cs.princeton.edu/~vivek/flash/](http://www.cs.princeton.edu/~vivek/flash/), consulté le 15/03/2010.
- [FON 07] R. Fontaine, P. Laurencot, A. Aussem, “Apprentissage des performances, surveillance en temps réel et contrôle d’admission d’un serveur Web utilisant les techniques neuronales”, 4th International Conference: Sciences of Electronic Technologies of Information and Telecommunications, March 25-29, 2007, Tunisie.

- [FUN 89] K. Funahashi (1989) “On the approximate realization of continuous mappings by neural networks”. *Neural Netw* 2(3):183–192
- [GEM 92] S. Geman, E. Bienenstock, R. Doursat, “Neural networks and the bias/variance dilemma”, *Neural Comp.* Vol. 4, No. 1, pp. 1–58, 1992.
- [GOO 10] “Classement des 1000 sites web les plus visités dans le monde”, <http://www.google.com/adplanner>, mars 2010.
- [GRO 04] B. Grone, A. Knopfel, R. Kugel, O. Schmidt, “The Apache Modelling Project”, Technical Report ISRN LUTFD2/TFRT--3239--SE, Hasso-Plattner-Institut for Software Systems Engineering, July 2004.
- [HEL 04] J. L. Hellerstein, Y. Diao, S. Parekh & D. M. Tilbury, “Feedback Control of Computing Systems”, , IEEE Press- John Wiley & Sons Inc., 2004.
- [HES 52] M. R. Hestenes, E. Stiefel, “Methods of conjugate gradients for solving Linear systems”. *J. of Res of National Bureau of Standards–B* 49:409–436, 1952.
- [HOP 82] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities”. *Proc. Nat. Acad. Sci.*, 79:2554–2558, 1982.
- [HOR 89] K. M. Hornik, M. Stinchcombe and H. White, “Multilayer feedforward networks are universal approximators”. *Neural Networks*, 2:359–366, 1989.
- [IIS 10] “Microsoft internet information services,” <http://www.microsoft.com/>, consulté le 15/03/2010.

- [JAI 91] R. Jain. “The Art of Computer Systems Performance Analysis”. John Wiley & Sons, 1991
- [JAN 88] P. Janssen, P. Stoica, T. Soderstrom, P. Eykhoff, “Model structure selection for multivariable systems by cross-validation”, *Int. J. Contr.*, 47:1737–1758, 1988.
- [KHT 10] “khttpd web server,” <http://www.fenrus.demon.nl/>, consulté le 13/03/2010.
- [KOB 78] H. Kobayashi, “Modelling and Analysis: An Introduction to System Performance Evaluation Methodology”, *The Systems Programming Series*, Addison Wesley, Reading, MA, 1978.
- [KOH 82] T. Kohonen, “Self-organized formation of topologically correct feature maps”, *Biol. Cybern.*, Vol. 43, pp. 59–69, 1982.
- [LEV 44] K. Levenberg, “A method for the resolution of certain non-linear problems in least-squares”, *Quart. Appl. Math.*, Vol. II, No. 2, pp. 164-168, 1944.
- [LIN 08] E. Lindegren, “preparing the Apache http server for feedback control application”. Master thesis, Lund University, February 2008.
- [LIN 10a] Linux manual page, <http://linux.die.net/man/2/listen>, consulté le 11/06/2010.
- [LIN 10b] Linux manual page, <http://linux.die.net/man/7/raw>, consulté le 12/06/2010.
- [LOO 97] C. Loosley, F. Douglas, and A. Mimo, “High-Performance Client/Server”, JohnWiley & Sons, 1997
- [MAH 05] Antoine Mahul, “Apprentissage de la qualité de service dans les réseaux multiservices : applications au routage optimal sous contraintes”, thèse de doctorat, université blaise pascal, France, 2005.
- [MAR 63] D. W. Marquardt, An algorithm for least-squares estimation of non-linear parameters, *J. Soc. Indust. Appl. Math.*, Vol. 11, No. 2, pp. 431-44, june 1963.
- [MAR 03] E. Marcus and H. Stern, “Blueprints for High Availability”, Wiley, Sept. 2003.

- [MAT 10] Site web officiel de Math Works 2, <http://www.mathworks.com>.
- [MEI 01] R. D. Van Der Mei, R. Hariharan, and P. K. Reeser, “web server performance modeling”, *Telecommunication systems*, vol. 16, no. 3,4, pp. 361-378, 2001
- [MCC 43] W. S. McCulloch, W. Pitts, “A logical calculus of the ideas immanent in nervous activity”, *Bull of Math Biophysics*, Vol. 5, pp. 115–133, 1943.
- [MIK 03] A. Mikael, C. Jianhua, K. Maria, N. Christian, “Performance modeling of an Apache web server with bursty arrival traffic”, *Proceedings of the International Conference on Internet Computing (IC’03)*, pp. 508-514, 23-26 juin 2003, Las Vegas, Nevada, USA.
- [MIN 69] M. L. Minsky, S. Papert, “Perceptrons”, MIT Press, Cambridge, MA, 1969.
- [MOL 93] M. F. Moller, “A scaled conjugate gradient algorithm for fast supervised learning”, *Neural Networks*, Vol. 6, N° 4, pp. 525-533, 1993.
- [MOS 98] D. Mosberger and T. Jin. “HTTPerf A Tool for Measuring Web Server Performance”. HP Research Labs. December 1998, volume 26 issue 3, *ACM Sigmetrics Performance Evaluation Review*
- [MOU 01] I. Mountain, “The Business Case for Disaster Recovery Planning : Calculating the Cost of Downtime”, 2001,
<http://www.ironmountain.com/dataprotection/resources/CostOfDowntimeIrnMtn.pdf>.
- [NAS 08] North American Systems International Inc. “The True Cost of Downtime”, 2008,
http://www.nasi.com/downtime_cost.php.

- [NER 93] O. Nerrand, P. Roussel-Ragot, L. Personnaz, G. Dreyfus, S. Marcos, “Neural networks and non-linear adaptive filtering: unifying concepts and new algorithms”, *Neural Computation*, Vol. 5, pp. 165-197, 1993.
- [NET 10] Netcraft Survey, disponible à l'adresse: <http://www.netcraft.com/survey/>.
- [PAL 06] N. Palluat, “Méthodologie de surveillance dynamique à l'aide des réseaux neuro-flous temporels”, Thèse de doctorat, Université de Franche-Comté, France, 2006.
- [PIN 87] F. J. Pineda, “Generalization of back-propagation to recurrent neural networks”, *Physical Rev. Lett.*, N° 59, pp. 2229–2232, 1987.
- [PLA 86] D. Plaut, S. Nowlan, G. E. Hinton, “Experiments on learning by back propagation”, Technical Report, Carnegie-Mellon University, 1986.
- [PUR 04] G. N. Purdy, “Linux Iptables: Pocket Reference”, O'Reilly Media, Inc., USA 2004.
- [REE 95] R. Reed, R. J. Marks II, S. Oh, “Similarities of error regularization, sigmoid gain scaling, target smoothing, and training with jitter”, *IEEE Trans. on Neural Networks*, Vol. 6, N° 3, pp. 529–538, 1995.
- [RUM 86] D. E. Rumelhart, G. E. Hinton, R. J. Williams, “Learning internal representations by error propagation”. In: Rumelhart DE, McClelland JL (Eds) *Parallel distributed processing: Explorations in the microstructure of cognition*, 1: Foundation, 318–362. MIT Press, Cambridge, 1986.
- [QIN 07] W. Qin, Q. Wang, “An LPV approximation for admission control of an internet web server: Identification and control”, *Control Engineering Practice*, No. 15, pp. 1457–1467, 2007.
- [RAF 05] F. Rafamantanantsoa, P. Laurençot, A. Aussem, “Analyse, Modélisation et Contrôle en Temps Réel des Performances d'un Serveur Web”, Rapport de Recherche LIMOS/RR-05-06, 10 mars 2005.

- [SAM 06] S. Samarasinghe, “Neural Networks for Applied Sciences and Engineering - From Fundamentals to Complex Pattern Recognition”, CRC Press, USA, 2006.
- [SCU 01] V. -M. Scuturici, “Utilisation efficace des serveurs Web en tant que serveurs vidéo pour des applications de vidéo à la demande”, Thèse de doctorat, Université Lumière Lyon 2, France, 2001.
- [STA 10] Statbrain Survey, <http://www.statbrain.com>.
- [SUN 08] Q. Sun, G. Dai, W. Pan, “LPV Model and Its Application in Web Server Performance Control”, College of Automation Northwestern Polytechnical University Xi’an, China 2008 International Conference on Computer Science and Software Engineering.
- [TUX 10] “Tux Web Server”, Reference manual : <http://www.redhat.com/docs/manuals/tux/TUX-2.1-Manual/>, consulté le 13/03/2010
- [UIT 94] E.800, “Terms and definitions related to quality of service and network performance including dependability. Recommendation”, ITU (UIT: Union Internationale des Télécommunications), 1994.
- [SPE 96] The Standard Performance Evaluation Corporation. SpecWeb96 at <http://www.spechbench.org/osg/web96>.
- [WAI 00] P. Wainwright, “Apache professionnel”, Editions Eyrolles, 2000.
- [WAI 04] P. Wainwright. “Pro Apache”, Third Edition. Apress, 2004.

- [XIA 05] C. Xiang, S. Q. Ding, T. H. Lee, “Geometrical interpretation and architecture selection of MLP”, IEEE Trans. on Neural Networks, N° 16, Vol. 1, pp. 84–96, 2005.
- [YAS 00] F. Yasuyuki, M. Masayuki, M. Hideo, “Performance modeling and evaluation of web server systems with proxy caching”, Ph.D thesis, Osaka University, 2000.
- [YEA 96] N. J. Yeager et R. E. McGrath, “Web Server Technology: The Advanced Guide for World Wide Web Information Providers”, Morgan Kaufmann Publishers, 1996.
- [YEU 99] K. H. Yeung, C. W. Szeto, “On the modeling of www request arrivals”, in Proceedings of the 1999 International Workshops on Parallel Processing, pp. 248–253, 1999.
- [ZEU 10] “Zeus web server,” <http://www.zeus.com>, consulté le 13/03/2010.

Les paramètres de réseau

Après avoir fait plusieurs itérations d'apprentissage on a obtenu le réseau de neurone suivant

Les couches :

Le réseau contient une couche d'entrée avec deux neurones et une couche de sortie avec un seul neurone et une fonction d'activation Hyperbolic tangent sigmoid (TANSIG). Le réseau contient une seule couche cachée avec 30 neurones et une fonction d'activation Log-sigmoid (LOGSIG).

Les poids :

Les poids entre la couche d'entrée et la couche cachée sont comme suit :

Entre le premier neurone de la couche d'entrée et les neurones de la couche cachée :

[9.2141 -2.3352 -33.3045 -2.9027 6.3259 9.0666 7.6826 4.9478 9.1466 -0.88597
-8.7072 14.0298 -15.7303 12.8171 12.7558 -13.6968 -5.5573 2.7878 18.3854
4.0172 1.2018 -9.8687 1.9108 -173.5812 48.7384 160.5468 5.6096 -146.296 8.3006
3.165]

Entre le deuxième neurone de la couche d'entrée et les neurones de la couche cachée :

[1.1589 -22.2391 8.2114 -0.80649 10.6613 4.704 13.4248 2.6412 15.6134 12.603
-15.4304 -19.0596 26.1395 -22.5621 -11.4501 19.2999 8.9473 11.4171 -25.6635
13.6916 -2.8576 -0.68502 -15.2602 -0.21415 -0.017424 0.18951 -15.4277 -0.15973
-15.3905 7.1271]

Entre les neurones de la couche cachée et le neurone de la couche de sortie :

[2.9933 -5.6459 -9.4057 -2.2222 -0.58494 -3.6971 3.6641 -0.28938 7.409 -0.98816 10.1416
11.2657 -6.0707 2.0353 -0.4084 16.7215 -0.35764 -20.3102 -1.498 13.6824 -6.3843 -11.3404
1.1867 -24.8883 24.2437 -69.0229 -2.2044 -125.2584 5.0375 13.6028]

Biais

Les poids entre le biais de la couche de la couche d'entre et les neurones de la couche cachée

[-8.5877 -10.4227 43.601 1.126 -10.8361 -6.4172 -6.2539 0.46655 -6.6932
8.6222 7.1092 5.2453 -6.754 12.5674 7.7751 -5.3378 0.46047 3.0835 8.2769
2.8658 1.0493 -13.2809 7.9152 -173.1568 52.7161 160.7925 11.5099 -148.2105
13.6352 18.9959]

Le poids entre le biais de la couche cachée et la couche de sortie :

[29.3302]