



الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLICUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE



MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET
DE LA RECHERCHE SCIENTIFIQUE

وزارة التعليم العالي و البحث العلمي

UNIVERSITE CONSTANTINE I

جامعة قسنطينة 1

FACULTE DES SCIENCES DE LA
TECHNOLOGIE
DEPARTEMENT DE GENIE CIVIL

كلية علوم التكنولوجيا
قسم : الهندسة المدنية

N° d'ordre :

Série :

Mémoire

Présenté en vue de l'obtention du diplôme de Magister

en Génie civil

Option : MATERIAUX

Présenté par : BENDAOUY YUCEF

Thème

***Prédiction Des Résistances Mécaniques Des Bétons à
Base Des Ciments Composés En Utilisant Les Réseaux
Neurones Artificiels***

Soutenue le : 07/07/ 2014 , Devant le Jury composé de:

Président	Dr CHABIL HOCINE	Pr Université de Constantine 1
Rapporteur	Dr GUETTECHE M NACER	Pr Université de Constantine 1
Examineur	Dr ROUABAH KAMEL	MC Université de Constantine 1
Examineur	Dr BEROUAL AHMED	MC Université de Constantine 1

JUIN 2014

REMERCIEMENTS

Je remercie **ALLAH** le tout puissant qui m'a guidé et qui m'a donné la force et la volonté de réaliser ce travail.

Je voudrai exprimer mes remerciements les plus sincères à mon directeur de thèse Monsieur **GUTTECHE MOHAMED NACER**, Professeur au Département de Génie Civil à l'université de Constantine 1, pour leur encadrement et leur expérience qu'il est su me faire partager.

Je voudrais exprimer toute ma reconnaissance à Monsieur **CHABIL H**, Professeur au Département de Génie Civil à l'université de Constantine1, pour l'honneur qu'il me fait en acceptant de présider le jury de cette thèse.

Je tiens également à remercier les membres de jury de soutenance, Monsieur **BEROUAL A** et Monsieur **ROUABAH K**, Maîtres de Conférences au Département de Génie Civil à l'université de Constantine 1, d'avoir accepté d'examiner la présente thèse.

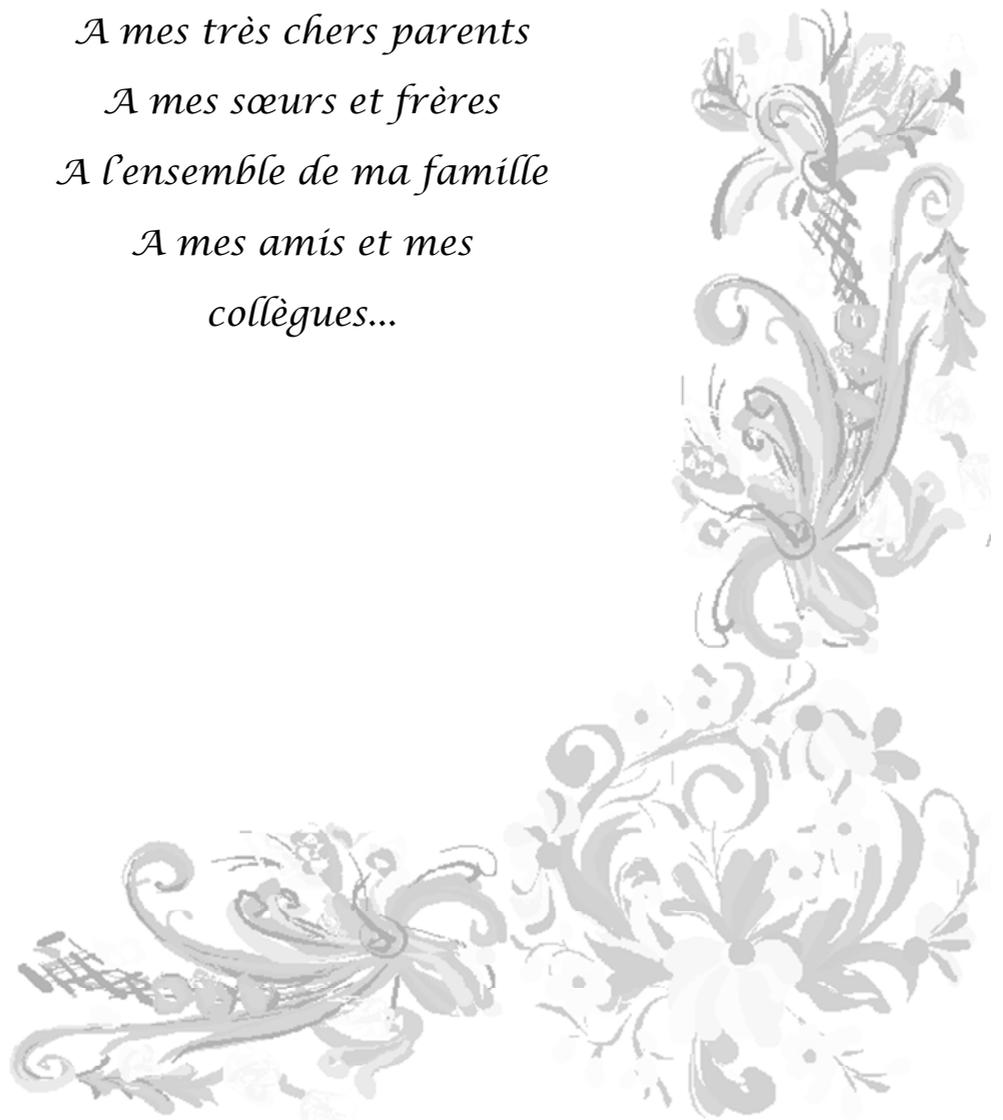
Mes remerciements et ma reconnaissance sont adressés envers mes collègues pour un temps précieux passé ensemble. Mes plus vifs remerciements vont également à tous mes amis avec qui j'ai partagé des moments inoubliables pendant mes études.

Enfin, toute ma gratitude, ma reconnaissance et mes très vifs remerciements à tous ceux qui ont contribué de près ou de loin et en particulier l'ensemble des enseignants du département de génie civil de Constantine1, à ma formation d'études de la graduation et de la post graduation.

BENDAOU D. Y

DÉDICACE

*Je dédie le présent travail
A mes très chers parents
A mes sœurs et frères
A l'ensemble de ma famille
A mes amis et mes
collègues...*



RESUME

Aujourd'hui, les matériaux hydrauliques (béton) sont les plus employés dans le monde pour les constructions civiles et industrielles, sans qu'on puisse pour autant toujours maîtriser ou prévoir leur comportement précis selon les diverses conditions d'emploi.

Au cours de la mise en place des matériaux hydrauliques, ils sont soumis à des évolutions physiques et chimiques, dépendent de la variation dans le dosage en ciment, le dosage en eau, la qualité des agrégats, la qualité du ciment et d'autres influences.

La présente étude a pour objet de montrer l'application d'une approche non paramétrique dite « Réseaux Neurones Artificiels » afin de prévoir l'évolution des variations dimensionnelles des compositions chimiques et minéralogiques de clinker à la résistance mécanique, l'expansion et la consistance.

L'utilisation de cette approche permet le développement de modèles pour leur prévision en utilisant un réseau multicouche à rétropropagation. Ils s'appuient également sur une grande base de données de résultats expérimentaux recueillis dans la littérature et sur un choix approprié des architectures et du processus d'apprentissage utilisés.

Mots clés :

Clinker, Pâte de Ciment, Composition chimique, Composition minéralogique, Réseaux de Neurones Artificiels, Béton, Résistance, Modèle, Apprentissage, Laitier.

ABSTARCT

Today, hydraulic materials (concrete) are the most used in the world for civil and industrial constructions, without being able to control or predict as always their specific behavior according to various conditions.

During the introduction of hydraulic materials, they are subjected to physical and chemical changes, the variation depends on the cement content, the water content, the quality of the aggregates, cement quality and other influences .

This study to show the application of a nonparametric approach called "Artificial Neural Networks" aims. to forecast the evolution of dimensional changes of chemical and mineralogical composition of clinker strength, expansion and consistency.

The use of this approach allows the development of models for their prediction using a multilayer network backpropagation. They are also based on a large database of experimental results collected in the literature and on an appropriate choice of architectures and processes of training utilized.

Key words :

Clinker, Cement Paste, Chemical Composition, Mineralogical Composition, Artificial Neural Network, Concrete, Strength, Model, Training , Slag.

:

اليوم، المواد الهيدروليكية (خرسانة) هي الأكثر استخداما في العالم في مجال المنشآت المدنية والصناعية, دون أن تكون هناك قدرة على السيطرة أو التنبؤ بسلوكها المحدد وفقا لشروط الاستعمال المختلفة.

خلال استعمال المواد الهيدروليكية, فهي تخضع لتغيرات فيزيائية وكيميائية, اعتمادا على التنوع في محتوى الإسمنت, محتوى المياه, نوعية الحصى, نوعية الاسمنت وتأثيرات أخرى.

من هذه الدراسة هو إظهار تطبيق طريقة لاجل التنبؤ, "شبكة العصبونات الاصطناعية" على تنبؤ مجال التغيرات في التركيبة الكيميائية و المعدنية للكلنكر على المقاومة, التمدد و التماسك.

استعمال هذه الطريقة يسمح بتطوير نماذج من أجل التنبؤ, نستعمل شبكة متعددة الطبقات ذات امتداد رجعي كما تستند على قاعدة من المعلومات لنتائج تجريبية موجودة في الأدبيات, و على الاختيار المناسب للتركيبات و طرق التعلم المستعملة.

كلمات مفتاحيه :

الكلنكر , عجينة الإسمنت , التركيب الكيميائي, التركيب المعدني, الشبكات العصبونية , خرسانة, مقاومة, نموذج, التعلم , البقايا.

RESUME

Aujourd'hui, les matériaux hydrauliques (béton) sont les plus employés dans le monde pour les constructions civiles et industrielles, sans qu'on puisse pour autant toujours maîtriser ou prévoir leur comportement précis selon les diverses conditions d'emploi.

Au cours de la mise en place des matériaux hydrauliques, ils sont soumis à des évolutions physiques et chimiques, dépendent de la variation dans le dosage en ciment, le dosage en eau, la qualité des agrégats, la qualité du ciment et d'autres influences.

La présente étude a pour objet de montrer l'application d'une approche non paramétrique dite « Réseaux Neurones Artificiels » afin de prévoir l'évolution des variations dimensionnelles des compositions chimiques et minéralogiques de clinker à la résistance mécanique, l'expansion et la consistance.

L'utilisation de cette approche permet le développement de modèles pour leur prévision en utilisant un réseau multicouche à rétropropagation. Ils s'appuient également sur une grande base de données de résultats expérimentaux recueillis dans la littérature et sur un choix approprié des architectures et du processus d'apprentissage utilisés.

Mots clés :

Clinker, Pâte de Ciment, Composition chimique, Composition minéralogique, Réseaux de Neurones Artificiels, Béton, Résistance, Modèle, Apprentissage, Laitier.

ABSTARCT

Today, hydraulic materials (concrete) are the most used in the world for civil and industrial constructions, without being able to control or predict as always their specific behavior according to various conditions.

During the introduction of hydraulic materials, they are subjected to physical and chemical changes, the variation depends on the cement content, the water content, the quality of the aggregates, cement quality and other influences .

This study to show the application of a nonparametric approach called "Artificial Neural Networks" aims. to forecast the evolution of dimensional changes of chemical and mineralogical composition of clinker strength, expansion and consistency.

The use of this approach allows the development of models for their prediction using a multilayer network backpropagation. They are also based on a large database of experimental results collected in the literature and on an appropriate choice of architectures and processes of training utilized.

Key words :

Clinker, Cement Paste, Chemical Composition, Mineralogical Composition, Artificial Neural Network, Concrete, Strength, Model, Training , Slag.

:

اليوم، المواد الهيدروليكية (خرسانة) هي الأكثر استخداما في العالم في مجال المنشآت المدنية والصناعية, دون أن تكون هناك قدرة على السيطرة أو التنبؤ بسلوكها المحدد وفقا لشروط الاستعمال المختلفة.

خلال استعمال المواد الهيدروليكية, فهي تخضع لتغيرات فيزيائية وكيميائية, اعتمادا على التنوع في محتوى الإسمنت, محتوى المياه, نوعية الحصى, نوعية الاسمنت وتأثيرات أخرى.

من هذه الدراسة هو إظهار تطبيق طريقة لالمعلمية تسمى "شبيكات العصبونات الاصطناعية" على تنبؤ مجال التغيرات في التركيبة الكيميائية و المعدنية للكلنكر على المقاومة, التمدد و التماسك.

استعمال هذه الطريقة يسمح بتطوير نماذج من أجل التنبؤ, نستعمل شبكة متعددة الطبقات ذات امتداد رجعي كما تستند على قاعدة من المعلومات لنتائج تجريبية موجودة في الأدبيات, و على الاختيار المناسب للتركيبات و طرق التعلم المستعملة.

كلمات مفتاحيه :

الكلنكر , عجينة الإسمنت , التركيب الكيميائي, التركيب المعدني, الشبيكات العصبونية , خرسانة, مقاومة, نموذج, التعلم , البقايا.

TABLE DES MATIERES.

Remerciements	<i>i</i>
Dédicace.....	<i>ii</i>
Résumé	<i>iii</i>
Table des matières.....	<i>vi</i>
Liste des symboles.....	<i>ix</i>
Liste des figures.....	<i>x</i>
List des tableaux	<i>xiii</i>
INTRODUCTION GENERALE	1
CHAPITRE 1. RESEAUX DE NEURONES.....	3
1.1 Introduction.....	4
1.2. Réseaux de neurones : définitions et propriétés	4
1.2.1. Neurone biologique.....	4
1.2.2. Neurone formel.....	6
1.3. Réseaux de neurones formels.....	7
1.3.1. Topologies.....	7
1.3.1.1 Réseaux de neurones statiques (réseaux non bouclés).....	8
1.3.1.2. Réseaux de neurones dynamiques ou réseaux bouclés (ou récurrents).....	9
1.3.2. Types d'apprentissage.....	9
1.3.3. Méthode de la rétro propagation du gradient (RPG).....	10
1.3.3.1. La théorie de l'algorithme d'optimisation de la rétro propagation du gradient	10
1.3.3.1.1. Réseau de neurone monocouche.....	10
1.3.3.1.2. Cas du Perceptron multicouche.....	17
1.3.3.1.3 Sommaire de l'algorithme dis de DELTA.....	19
1.3.3.1.4 Règle de « DELTA généralisé ».....	20
1.3.3.1.5. Techniques de réglage du pas	20
1.3.3.1.6. Résilient back propagation.....	20
1.3.3.1.7. Méthode d'apprentissage par algorithme du gradient conjugué.....	21
1.3.3.1.8 Les méthodes de gradient du second ordre.....	21
1.4 Conclusion.....	23
CHAPITRE 2. HISTORIQUE ET APPLICATIONS DES RESEAUX DE	
NEURONES.....	24
2.1. Historique.....	25
2.1.1. Les premiers succès.....	27
2.1.2. L'ombre.....	29
2.1.3. Les renouveaux.....	32

2.1.4. La levée des limitations.....	34
2.2. Apprentissage.....	37
2.2.1. Rétropropagation du gradient	37
2.2.1.1. Historique.....	38
2.2.1.2. Utilisation au sein d'un apprentissage supervisé	38
2.3. Les Réseaux De Neurones Appliqués Aux Génie Civil	40
2.3.1. Application des RNA en analyse des structures	41
2.3.2. Application des RNA en géotechnique.....	41
2.3.3. Application des RNA en technologie des bétons.....	41
2.3.4. Composition des bétons.....	42
2.3.5. Hydratation du ciment.....	43
2.3.6. Retrait et fluage des bétons	43
2.3.7. Durabilité et autres.....	44
2.4. Avantages des RNA.....	44
CHAPITRE 3. MODELISATION PAR RESEAUX DE NEURONES ARTIFICIELS	
(RNA).....	45
3.1. Introduction.....	46
3.2. Réseaux Neurones Multicouche à Rétropropagation.....	46
3.2.1. Définition.....	46
3.2.2. Structure et fonctionnement.....	47
3.2.3. Phase d'apprentissage.....	49
3.2.3.1. Introduction des données.....	49
3.2.3.2. Calcul de l'erreur.....	50
3.2.3.3. Modification des poids synaptiques liants la couche de sortie.....	50
3.2.3.4. Modification des poids synaptiques liants la couche cachée.....	51
3.2.4. Phase de reconnaissance (test) et de validation.....	52
3.2.5. Performances du réseau de neurones multicouche à rétropropagation.....	52
CHAPITRE 4. MODELES RESEAUX DE NEURONES SUR LE MORTIER.....	
4.1. Introduction.....	54
4.2. Méthodologie et mise en œuvre.....	54
4.3. Langage de programmation utilisé.....	56
4.4. Elaboration des modèles RNA.....	58
4.4.1. Introduction.....	58
4.4.2. Modèle Global du Résistance a la compression.....	58
i. Analyse Chimique.....	58
ii. Analyse minéralogique.....	63

iii. Etude paramétrique.....	67
a) Influence des alcalis(K ₂ O)	67
b) Influence de MgO	68
c) Influence du C3S	69
d) Influence du C2S	70
e) Influence du C3A	70
4.4.3. Modèle Global de l'expansion.....	71
i. Analyse Chimique.....	71
ii. Analyse minéralogique.....	74
iii. Etude paramétrique.....	77
a) Influence de SiO ₂	77
b) Influence des alcalis (K ₂ O)	78
c) Influence de MgO	79
4.4.4. Modèle Global de la consistance.....	79
i. Analyse Chimique.....	79
ii. Analyse minéralogique.....	83
iii. Etude paramétrique.....	86
a) Influence de SiO ₂	86
b) Influence de Fe ₂ O ₃	87
c) Influence des alcalis Na ₂ O.....	88
d) Influence des alcalis (K ₂ O)	88
4.5. Conclusion.....	89
CHAPITRE 5. MODELES RESEAUX DE NEURONES SUR LE BETON.....	90
5.1 Le béton hydraulique.....	91
5.2 les constituants utilisés.....	91
5.3 Etude expérimentales de la composition du béton et méthodologie de composition.....	92
5.4 Les résultats expérimentales.....	93
5.5 Elaboration des modèles RNA.....	94
5.5.1 Introduction.....	94
5.5.2 Modèle Global du Résistance a la compression.....	94
5.5.3 Modèle Global du Résistance a la traction.....	97
5.5.4 l'influence de la variation de la teneur en laitier.....	100
5.6 Conclusion.....	101
CONCLUSION GENERALE	103
REFERENCES BIBLIOGRAPHIQUE	105
ANEXXE	108

LISTE DE NOTATION ET ABRÉVIATIONS

<i>CEM II/ A 32,5 R</i>	Ciment Portland à prise rapide dont la résistance caractéristique minimum en compression à 28 jours est égale à 32,5 MPa R
<i>C₄AF</i>	Ferro aluminate tétra calcique qui confère sa couleur grise au ciment gris.
<i>SIO₂</i>	Silice
<i>FE₂O₃</i>	Oxyde Ferrique
<i>AL₂O₃</i>	Oxyde Alumine
<i>K₂O</i>	Oxyde Potassium
<i>NA₂O</i>	Oxyde Sodium
<i>C₃A</i>	Aluminate tricalcique, une des phases pures du ciment
<i>C₃S</i>	Silicate tricalcique, la phase pure majoritaire du ciment
<i>C₂S</i>	Silicate bicalcique, une des phases pures du ciment
<i>Ca(OH)₂</i>	Hydroxyde de calcium
<i>C-S-H</i>	Silicate de calcium hydraté, produit issu de l'hydratation du C ₃ S
<i>E/C</i>	Rapport eau sur ciment
<i>MgO</i>	Oxyde de magnésium
<i>SO₃</i>	Groupe sulfonâtes
<i>RNA</i>	Réseaux De neurone Artificiel
<i>RN</i>	Réseaux De neurone
<i>PMC</i>	Perceptron Multicouches
<i>PRG</i>	Retro propagation du gradient
<i>L-M</i>	Levenderg –marquardt
<i>MSE</i>	L'erreur quadratique moyenne (Mean Square Error)
<i>SSE</i>	La somme de l'erreur quadratique (Sum Of Squared Error)
<i>R²</i>	Coefficient de détermination
<i>MRNGPR</i>	Modèle Réseaux De Neurone Globale De La Prévision Du Resistance
<i>MRNGPC</i>	Modèle Réseaux De Neurone Globale De La Prévision Du Consistance
<i>MRNGPE</i>	Modèle Réseaux De Neurone Globale De La Prévision Du Expansion
<i>MPa</i>	Mega Pascal
<i>Exp</i>	Expansion
<i>CON</i>	Consistance

LISTE DES FIGURES

Figure (1-1) Exemple de neurone biologique (Jayet., 2002).....	5
Figure (1-2) Structure générale du neurone formel	7
Figure (1-3) Topologies des RNA (Sorin et al., 2001).....	8
Figure (1-4) Perceptron non récurrent ou feed-forward à une sortie	10
Figure (1-5) Gradient de l'erreur totale.....	12
Figure (1-6) Perceptron multicouche	14
Figure (1-7) Connexions entre les neurones d'un PMC	14
Figure (1-8) Procédure d'élaboration d'un modèle neuronal	19
Figure (1-9) Nombre d'article publiés qui utilisent dans leur titre le terme (neurones) dans le journal of computing in civil engineering (flood 2006)	23
Figure (2-1) modèle de l'activité perspective.....	27
Figure (2-2) forme basique d'une carte topologique de Kohonen.....	30
Figure (2-3) organisation des cellules de modèle de Kohnen.....	30
Figure (2-4) modèle de Hopfield.....	33
Figure (2-5) Forme des machines de Boltzmann.....	35
Figure (2-6) Machine de Boltzmann restreinte.....	35
Figure (3-1) Structure d'un neurone artificiel	46
Figure (3-2) Topologie d'un réseau de neurones multicouche à rétropropagation	46
Figure (3-3) Organigramme de Rétropropagation (Hecht-Nielsen 1989).	51
Figure (4-1) Méthodologie de développement du Réseau de neurones artificiels (RNA).	55
Figure (4-2) Panneau principal de l'application NETWORK DATA MANAGER.	56
Figure (4-3) Architecture adoptée pour la Résistance en fonction de la composition chimique de clinker.....	60
Figure (4-4) Apprentissage du modèle MRNGPR	62
Figure (4-5) : Performance du Modèle Global MRNGPR(la composition chimique).....	63
Figure (4-6): Architecture adoptée pour la Résistance.....	65
Figure (4-7) Apprentissage de modèle MRNGPR (la composition minéralogique)	66
Figure (4-8) Performance du Modèle Global MRNGPR(la composition minéralogique)..	67

Figure (4-9) Influence des alcalis sur les résistances à la compression	67
Figure (4-10) Influence de MgO sur les résistances à la compression	68
Figure (4-11) Influence du C ³ S sur les résistances à la compression d'un pate du clinker	69
Figure (4-12) Influence du C ₂ S sur les résistances à la compression	70
Figure (4-13) Influence du C ₃ A sur les résistances à la compression	70
Figure (4-14) : Architecture adoptée pour l'expansion en fonction de la composition chimique de clinker.....	72
Figure (4-15) Performance du Modèle Global MRNGPE(la composition chimique).	73
Figure (4-16) Apprentissage du modèle MRNGPE	73
Figure (4-17) : Architecture adoptée pour l'expansion en fonction de la composition minéralogique de clinker.....	75
Figure (4-18) Performance du Modèle Global MRNGPE(la composition minéralogique).....	76
Figure (4-19) Apprentissage du modèle MRNGPE (la composition minéralogique).....	76
Figure (4-20) Influence de SiO ₂ sur l'expansion	77
Figure (4-21) Influence des alcalis sur l'expansion	78
Figure (4-22) Influence de MgO sur l'expansion	79
Figure (4-23) Architecture adoptée pour la consistance en fonction de la composition chimique de clinker.....	81
Figure (4-24) Apprentissage du modèle MRNGPC (composition chimique).....	82
Figure (4-25) Performance du Modèle Global MRNGPC(composition chimique).	82
Figure (4-26) : Architecture adoptée pour la consistance en fonction de la composition minéralogique de clinker.	84
Figure (4-27) Apprentissage du modèle MRNGPC (composition chimique).	85
Figure (4-28) Performance du Modèle Global MRNGPC(composition chimique).	85
Figure (4-29) Influence de SiO ₂ sur la consistance	86
Figure (4-30) Influence de Fe ₂ O ₃ sur la consistance	87
Figure (4-31) Influence des alcalis sur la consistance.	88
Figure (4-32) Influence des alcalis sur la consistance	88
Figure (5-1) Architecture adoptée pour la Résistance à la compression du béton en fonction du pourcentage de laitier	92
Figure (5-3) Performance du Modèle Global MRNGPRc.....	94

Figure (5-4) Architecture adoptée pour la résistance à la traction du béton en fonction du pourcentage de laitier	95
Figure (5-5) Apprentissage du modèle MRNGPRt.....	96
Figure (5-6) Performance du Modèle Global MRNGPRt.....	97
Figure (5-7) Influence de pourcentage de laitier sur les résistances à la compression.....	97
Figure (5-8) Influence de pourcentage de laitier sur les résistances à la traction.....	98

LISTE DES TABLEAUX

Tableau (1-1) Formes et graphes de quelques fonctions d'activation usuelles.....	6
Tableau (1-2) Glossaire réseaux de neurones / statistiques (Thiria et al. 1997).....	2
Tableau (4-1): Indicateurs utilisés – plage de variation	58
Tableau (4-2): Comparaison entre les algorithmes d'apprentissage.....	59
Tableau (4-3): Résumé de l'effet du nombre de couches cachées.....	59
Tableau (4-4): Architecture et paramètres de MRNGPR	61
Tableau (4-5) : performance de modèle MRNGPR.....	61
Tableau (4-6): Indicateurs utilisés – plage de variation.....	63
Tableau (4-7): Résumé de l'effet du nombre de couches cachées.....	64
Tableau (4-8): performance de modèle MRNGPR (la composition minéralogique).....	65
Tableau (4-9): Indicateurs utilisés – plage de variation.....	71
Tableau 4-10 : performance de modèle MRNGPE.....	72
Tableau (4-11): Indicateurs utilisés – plage de variation.....	74
Tableau 4-12 : performance de modèle MRNGPE.....	75
Tableau (4-13): Indicateurs utilisés – plage de variation.....	80
Tableau (4-14): performance de modèle MRNGPC.....	81
Tableau (4-15): Indicateurs utilisés – plage de variation.....	83
Tableau (4-16) : performance de modèle MRNGPC	84
Tableau (5-1) : résultats des résistance en compression et traction des différents pourcentage de laitier.....	90
Tableau (5-2): Indicateurs utilisés – plage de variation.....	91
Tableau (5-3) : performance de modèle MRNGPRc.....	93
Tableau (5-4): Indicateurs utilisés – plage de variation.....	94
Tableau (5-5) : performance de modèle MRNGPRt.....	96

INTRODUCTION GENERALE

1. Généralités

L'utilisation du matériau béton a peu évolué pendant des siècles jusqu'à l'apparition du béton armé, inventé en 1867 par Joseph Monnier, et du béton précontraint dont la technique fut définitivement mise au point vers 1930 par Eugène Freyssinet.

Aujourd'hui, les matériaux hydrauliques (béton) sont les plus employés dans le monde pour les constructions civiles et industrielles, sans qu'on puisse pour autant toujours maîtriser ou prévoir leur comportement précis selon les diverses conditions d'emploi. Ceci explique la multitude de recherches et d'approches pour tenter d'améliorer la connaissance de ce matériau. Ces recherches portent aussi bien sur des aspects économiques, sociaux, esthétiques que sur des aspects physico-chimiques et mécaniques.

Au cours de la mise en place des matériaux hydrauliques, ils sont soumis à des évolutions physiques et chimiques, dépendent de la variation dans le dosage en ciment, le dosage en eau, la qualité des agrégats, la finesse du ciment et d'autres variations liées aux différentes opérations telles que le bétonnage, le malaxage, vibration etc ...

la variabilité de la matière première de clinker et ciment peuvent conduire à des variations importantes sur la qualité du béton.

2. Problématique

La prévision de la résistance des bétons d'aujourd'hui est devenue une préoccupation majeure.

Les bétons actuels possèdent une composition caractérisée par la présence de certains ajouts minéraux pour améliorer ses performances.

L'objectif de cette étude est de modéliser la contribution des ajouts minéraux à la résistance en compression des bétons afin de pouvoir, à terme intégrer ces matériaux dans un logiciel d'aide à la formulation (un organigramme de prévision).

Cette modélisation se fait en utilisant les réseaux neurones artificiels, en se basant sur un modèle de résistance déjà existant, et grâce à des données expérimentales provenant d'une part, et de la littérature d'autre part, d'une campagne d'essais spécifique réalisée avec des formules de mortiers et bétons (si c'est possible).

Les réseaux neurones artificielles ont la capacité de classification de connaissance, de simulation et de prise de décision qui leur ont conféré un champ d'application en ingénierie, notamment dans le domaine des matériaux en génie civil.

3. Objectif du travail:

- L'objectif premier de la présente étude est de modéliser la contribution des ajouts minéraux à les caractéristiques physiques et mécanique du mortier à fin de pouvoir, à terme intégrer ces matériaux dans un logiciel .
- Une modélisation se fait en utilisant les réseaux neurones artificiels, en se basant sur un modèle de résistance déjà existant.
- L'avantage de l'utilisation de réseaux neurones en ingénierie, notamment dans le domaine des matériaux en génie civil.

Le présent travail est structuré comme suite:

Une introduction générale qui inclut la problématique qui encouragé cette recherche et leurs objectifs.

La première partie se compose de deux chapitres. Le premier chapitre présente les définitions et différents types des réseaux neurones et différents types d'algorithme d'apprentissage.

Dans le deuxième chapitre, nous présentons les étapes de développement des modèles de réseaux neurones , nous donnons aussi l'application des réseaux neurones aux génie civil et les avantages des réseaux neurones artificiels .

La deuxième partie comporte trois chapitres.

Dans le troisième chapitre, nous modélisons par Réseaux Neurones Multicouche à Rétropropagation de gradient de l'erreur .

Dans le quatrième chapitre, nous modélisons l'évolution des résistances mécanique et consistance et l'expansion sur pâte du clinker et des mortier normalisés à base d'un ciment CPJ-CEMII/A 32.5 par l'application de la technique des réseaux de neurones artificiels .

Enfin, dans le cinquième chapitre , nous modélisons l'évolution des résistances mécanique des bétons à base d'un ciment de laitier par l'application de la technique des réseaux de neurones artificiels .

Le travail est clôturé par une Conclusion générale .

1. RESEAUX DE NEURONES

1.1 Introduction

Lors de l'émergence d'une nouvelle technique, l'ingénieur se demande naturellement en quoi elle peut lui être utile. S'elle est dotée d'un nom plus métaphorique que scientifique, ce qui est évidemment le cas pour les réseaux de neurones, dont le nom est lié fortement à la biologie.

Alors techniquement parlant, les réseaux de neurones artificiels (RNA) sont de Nature purement mathématique et statistique ; leurs applications se situent dans des domaines qui n'ont généralement aucun rapport avec la neurobiologie.

Le domaine de calcul par réseaux de neurones a mûri au cours de la dernière décennie et a trouvé de nombreuses applications industrielles. Les réseaux de neurones sont maintenant couramment utilisés dans les processus de contrôle, de fabrication, de contrôle de la qualité, de la conception du produit, d'analyse financière, de la détection des fraudes, de l'approbation du prêt, de la reconnaissance vocale et d'exploration de données.

1.2. Réseaux de neurones : définitions et propriétés

Les réseaux de neurones artificiels ou réseaux connexionnistes sont fondés sur des modèles qui tentent de mimer les cellules du cerveau humain et leurs interconnexions. Le but, d'un point de vue global, est d'exécuter des calculs complexes et de trouver, par apprentissage, une relation non linéaire entre des données numériques et des paramètres.

1.2.1. Neurone biologique

Le cerveau humain possède deux hémisphères latérales reliées par le corps calleux et d'autres ponts axonaux, il pèse moins de deux kilogrammes et contient mille milliards de cellules, dont 100 milliards sont des neurones constitués en réseaux.

Le neurone biologique (figure. 1.1) est une cellule vivante spécialisée dans le traitement des signaux électriques. Les neurones sont reliés entre eux par des liaisons appelées axones. Ces axones vont eux-mêmes jouer un rôle important dans le comportement logique de l'ensemble.

Ils conduisent les signaux électriques de la sortie d'un neurone vers l'entrée (synapse) d'un autre neurone. Les neurones font une sommation des signaux reçus en entrée et en fonction du résultat obtenu vont fournir un courant en sortie [1] .

Les neurones sont des cellules nerveuses décomposables en 4 parties principales (figure. 1.1).

Les dendrites, sur lesquelles les autres cellules entrent en contact synaptique : c'est par les dendrites que se fait la réception des signaux. Le corps de la cellule ou noyau, c'est l'unité de traitement. L'axone est la partie où passent les messages accumulés dans le corps de la cellule. Enfin, à la sortie du neurone on trouve les synapses, par lesquelles la cellule communique avec d'autres neurones, ce sont des points de connexion par où passent les signaux de la cellule.

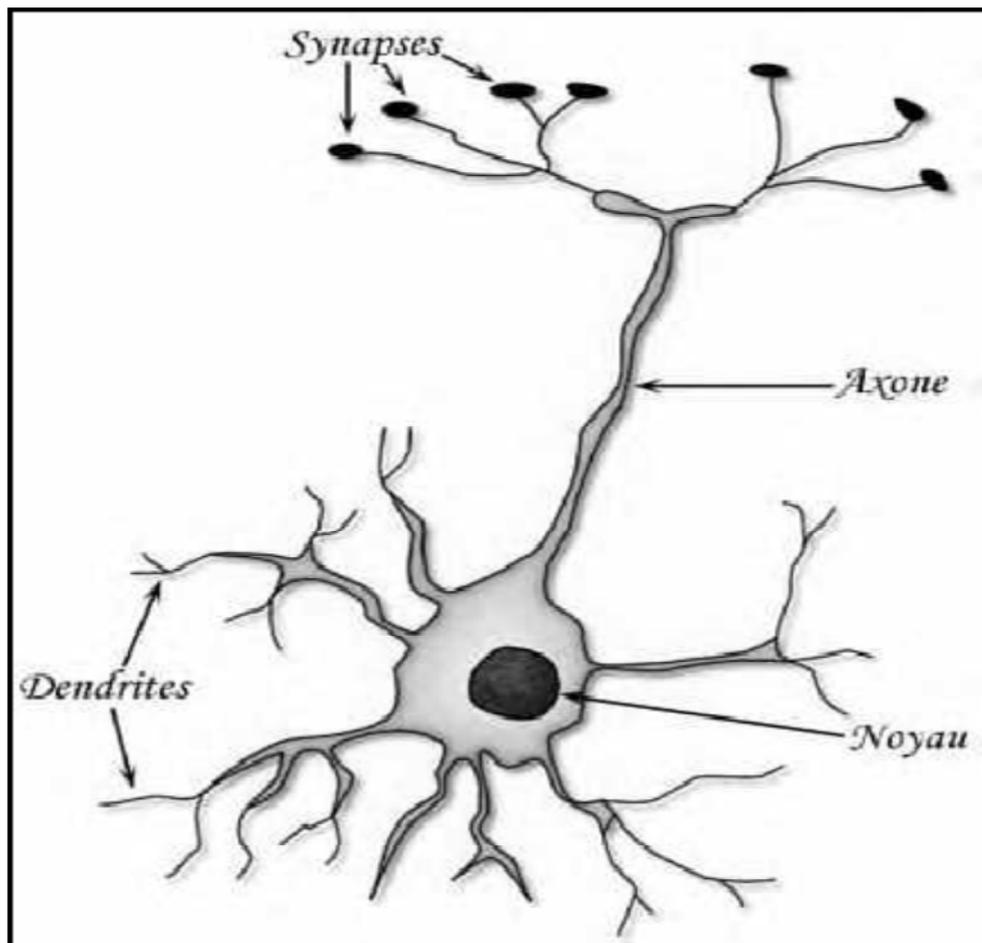


Figure (1-1) Exemple de neurone biologique [2] .

Un neurone stimulé envoie des impulsions électriques ou potentielles d'action, à d'autres neurones. Ces impulsions se propagent le long de l'axone unique de la cellule. Au point de contact entre neurones, les synapses, ces impulsions sont converties en signaux chimiques.

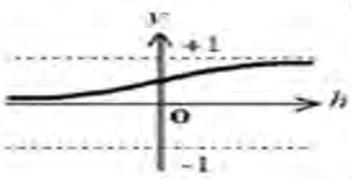
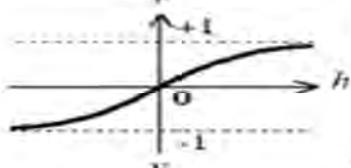
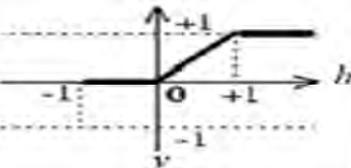
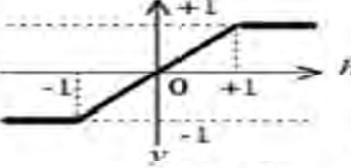
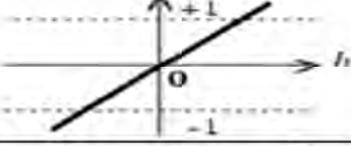
Quand l'accumulation des excitations atteint un certain seuil d'activation, le neurone engendre un potentiel d'action, pendant une durée de 1ms. Le neurone émettant le signal est appelé le neurone pré-synaptique et celui recevant ce signal, neurone post-synaptique.

1.2.2. Neurone formel

Par définition un neurone formel (figure. 1.2) est une fonction algébrique non linéaire, paramétrée, à valeurs bornées [3] . Il peut être caractérisé par :

- a) La nature de ses entrées ($x_1, x_2, \dots, x_i, \dots, x_n$), qui peuvent être les entrées du réseau ou les sorties d'autres neurones du même réseau,
- b) La fonction d'entrée totale définissant le prétraitement (combinaison linéaire) effectué sur les entrées comme $\sum W_i \cdot x_i + b$, dont W_i est le poids synaptique attaché à l'entrée i et le b désigne le seuil d'activation (biais),
- c) Sa fonction d'activation, ou d'état f , définissant l'état interne du neurone en fonction de son entrée totale. Cette fonction peut prendre plusieurs formes (Tableau 1.1 : liste non exclusive),
- d) Sa fonction de sortie calculant la sortie du neurone en fonction de son état d'activation.

Tableau (1-1) Formes et graphes de quelques fonctions d'activation usuelles.

Type de fonction d'activation	formules	Graphes
Logistique-sigmoïde	$y = \frac{1}{1 + e^{-h}}$	
Tanh-sigmoïde	$y = \frac{e^h - e^{-h}}{e^h + e^{-h}}$	
Linéaire Logistique-Sigmoïde	$y = \begin{cases} 0 & h < 0 \\ 1 & h > 1 \\ h & \text{sinon} \end{cases}$	
Linéaire Tanh-sigmoïde	$y = \begin{cases} -1 & h < -1 \\ 1 & h > 1 \\ h & \text{sinon} \end{cases}$	
Linéaire	$y = h$	

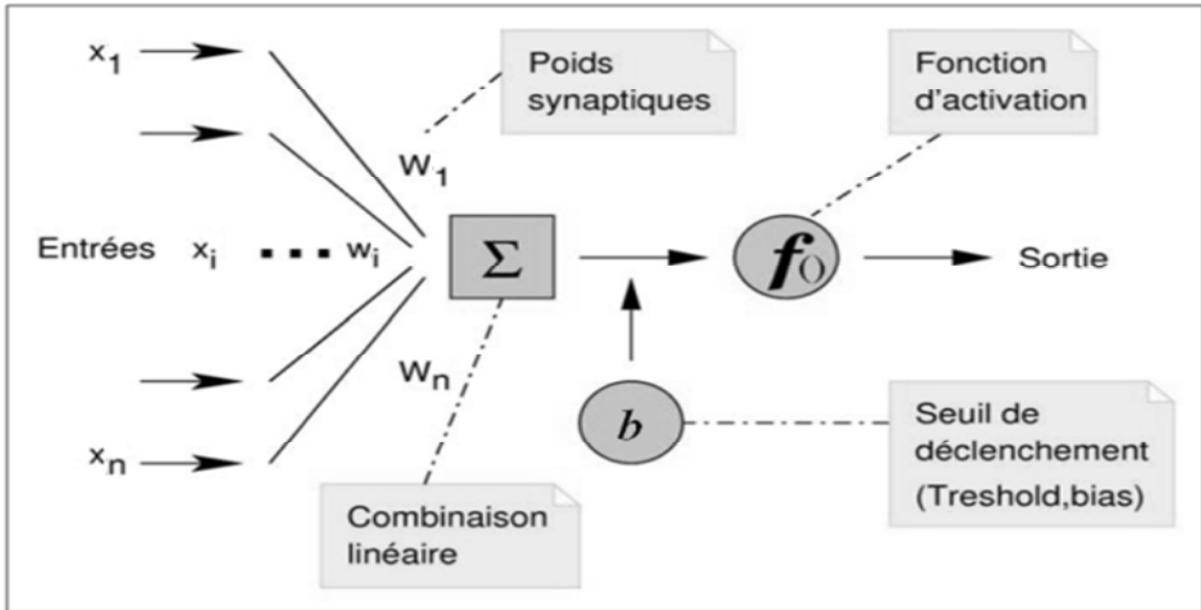


Figure (1-2) Structure générale du neurone formel

1.3. Réseaux de neurones formels

Les réseaux de neurones artificiels regroupent en réseaux un certain nombre de neurones formels connectés entre eux de diverses manières. Un réseau est défini par sa topologie, qui représente le type de connexion existant entre les divers neurones du réseau, par la fonction d'activation qui le caractérise et par les méthodes d'apprentissage utilisées pour trouver une relation non linéaire optimale par approximation entre les variables d'entrées et de sorties.

1.3.1. Topologies

Les neurones sont connectés entre eux de diverses manières. (a) Réseaux multicouche à connexions Totales, (b) à connexions locales, (c) réseau à connexions complexes et (d) réseau multicouche à connexions récurrentes (dynamique) (Figure. 1.3.).

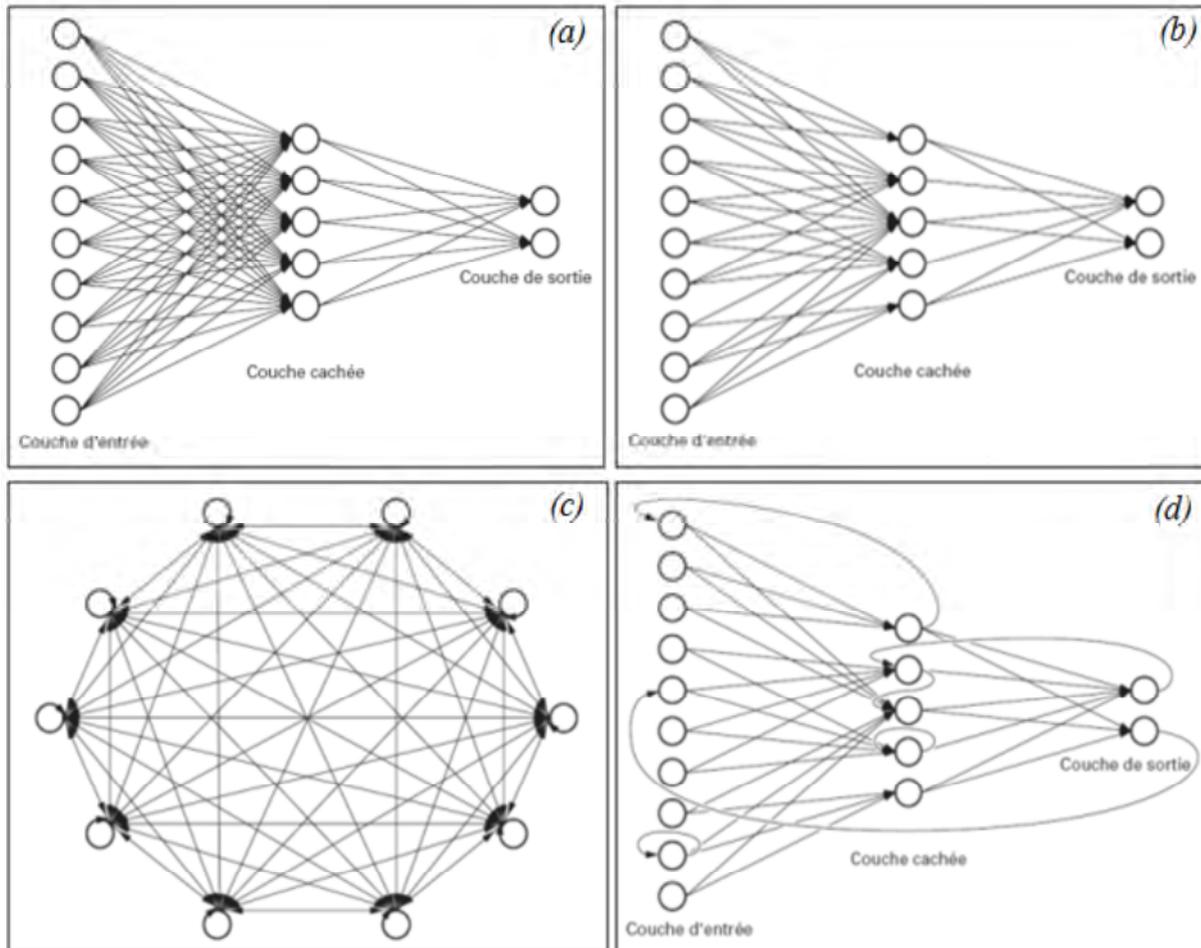


Figure (1-3) Topologies des RNA [1] .

De la figure. 1.3 on peut distinguer deux familles de réseaux de neurones : non bouclés ou statiques (a) et (b) et bouclés (dynamiques) (c) et (d).

1.3.1.1 Réseaux de neurones statiques (réseaux non bouclés)

L'exemple le plus simple de réseau de neurones statique est donné par le perceptron multicouche chaque couche contenant un ou plusieurs neurones. Afin d'illustrer ces propos, voici le rôle de chacune des couches dans un perceptron à trois couches.

La première couche est appelée couche d'entrée. Elle recevra les données source que l'on veut utiliser pour l'analyse. Sa taille est donc directement déterminée par le nombre de variables d'entrées.

La deuxième couche est la couche cachée. Dans cette couche, les fonctions d'activation sont en général non linéaires. Le choix de sa taille (nombre de neurones) n'est pas automatique et doit être ajusté. Il sera souvent préférable pour obtenir la taille optimale, d'essayer le plus de tailles possibles.

La troisième couche est appelée couche de sortie. Elle donne le résultat obtenu après compilation par le réseau des données entrée dans la première couche. Sa taille est directement déterminée par le nombre de variables dont on a besoin en sortie. C'est le réseau statique non bouclé qu'on va utiliser dans cette étude, car les paramètres recherchés dans le présent projet de recherche sont indépendants du temps .

1.3.1.2. Réseaux de neurones dynamiques ou réseaux bouclés (ou récurrents)

L'architecture la plus générale, pour un réseau de neurones, est celle des « réseaux bouclés », dont le graphe des connexions est cyclique : lorsque l'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 un tel chemin est désigné sous le terme de « cycle ». La sortie d'un neurone du réseau peut donc être fonction d'elle même ; ceci n'est évidemment concevable que si la notion de temps est explicitement prise en considération [3] .

1.3.2. Types d'apprentissage

L'objectif de la phase d'apprentissage des RNA est de trouver, parmi toutes les fonctions paramétrées par les poids synaptiques, celle qui s'approche le plus possible de l'optimum. L'apprentissage consiste, donc à minimiser une fonction de coût à l'aide des algorithmes d'optimisation. L'un d'eux est décrit dans la section suivante.

Il existe deux types d'apprentissage : l'apprentissage supervisé et l'apprentissage non supervisé. Le superviseur, ou professeur, fournit au réseau des couples d'entrée-sortie. Il fait apprendre au réseau l'ensemble de ces couples, par une méthode d'apprentissage, comme la rétro-propagation du gradient de l'erreur, en comparant pour chacun d'entre eux la sortie effective du réseau et la sortie désirée. L'apprentissage est terminé lorsque tous les couples entrée-sortie sont reconnus par le réseau. Ce type d'apprentissage se retrouve, entre autres, dans le perceptron. Le deuxième type c'est l'apprentissage non supervisé. Cet apprentissage consiste à détecter automatiquement des régularités qui figurent dans les exemples présentés et à modifier les poids des connexions pour que les exemples ayant les mêmes caractéristiques de régularité provoquent la même sortie. Les réseaux auto-organiseurs de Kohonen (1981) sont les réseaux à apprentissage non supervisé les plus connus [4] .

La plupart des algorithmes d'apprentissage des réseaux de neurones formels sont des algorithmes d'optimisation (méthode supervisée). La méthode d'optimisation la plus célèbre et la plus utilisée en analyse neuronale est la rétro-propagation du gradient de l'erreur.

1.3.3. Méthode de la rétro propagation du gradient (RPG)

1.3.3.1. La théorie de l'algorithme d'optimisation de la rétro propagation du gradient

L'apprentissage se fait le plus souvent avec l'algorithme de la rétro-propagation du gradient de l'erreur (RPG). Il a été créé en généralisant les règles d'apprentissage de Widrow-Hoff [5] aux réseaux mono et multicouches à fonction de transfert non linéaire. Le but est d'ajuster les poids synaptiques et les biais pour lesquels la fonction de coût des moindres carrés, calculée sur les points de l'ensemble d'apprentissage, soit minimale.

A chaque couple entrée/sortie, une erreur est calculée, le gradient de l'erreur est déterminé. Ce gradient est la dérivée partielle de l'erreur par rapport au poids synaptique. Par la RPG, les poids et les biais sont modifiés sur le réseau. Soit en ligne (les poids sont modifiés après chaque exemple de l'ensemble d'apprentissage) ou en paquet (les poids sont modifiés après que tous les exemples aient défilé). On réitère ces calculs jusqu'à l'obtention du critère d'arrêt.

C'est un algorithme utilisé pour l'apprentissage de fonction, la reconnaissance de formes et la classification. Voyons, le cas le plus simple, celui du Perceptron.

1.3.3.1.1. Réseau de neurone monocouche : le perceptron

Le perceptron peut être considéré comme le premier des RNA. Il fut mis au point dans les années cinquante par Rosenblatt [6]. Comme son nom l'indique, le perceptron se voulait un modèle de l'activité perspective. Il se compose d'une rétine et d'une couche qui donne la réponse correspondant à la simulation donnée en entrée. Il existe 2 types de perceptrons : les perceptrons (feed-forward) et les perceptrons récurrents.

Les perceptrons récurrents sont ceux qui alimentent leurs entrées avec leurs sorties, par opposition aux perceptrons feed-forward.)

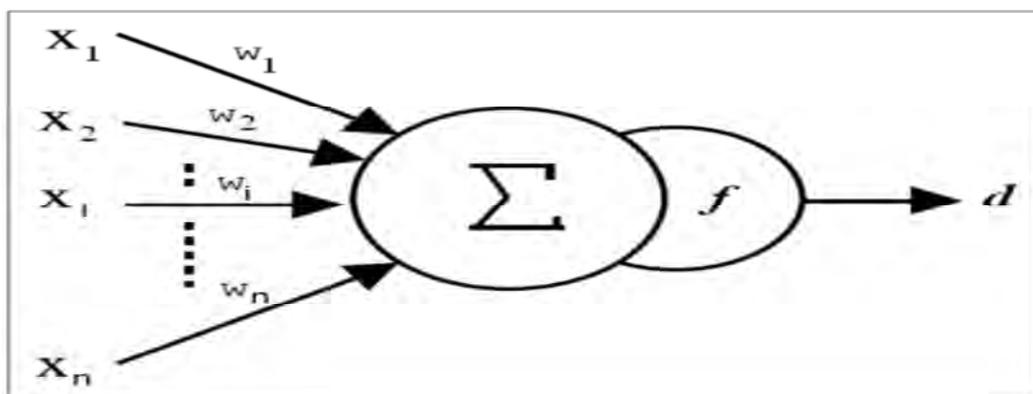


Figure (1-4) Perceptron non récurrent ou feed-forward à une sortie

L'idée générale de la technique de la rétropropagation du gradient est de chercher le minimum de la fonction du coût (sortie désirée-sortie calculée) qui dépend des poids synaptique W_i et ce par itération successive.

L'algorithme se déroule comme suit :

1. Initialiser les poids W_i de manière aléatoire;
2. Propager le signal de valeurs de l'entrée vers la couche de sortie ;
3. Calculer le gradient de l'erreur notée $E = \frac{\partial E}{\partial W_i}$ (figure. 1.5);

La fonction E à minimiser est la somme de l'erreur quadratique moyenne de moindre carrés définie par :

$$E = \frac{1}{2} (d - y(h))^2$$

Avec : $y(h) = f\left(\sum_{i=1}^n W_i \cdot x_i\right)$

$$h = \sum_{i=1}^n W_i \cdot x_i$$

y : la sortie calculée par le RNA

h : la somme pondérée des entrées du neurone de la couche de sortie.

f : la fonction d'activation

d : la sortie désirée

x_i : Variable d'entrée de la composante i

On peut écrire E sous la forme $E = \frac{1}{2} \delta^2$ $\delta = (d - y(h))$

4. Corriger W_i en direction inverse du gradient (la rétro-propagation) :

$$W_i \leftarrow W_i - \delta \cdot W_i$$

L'opérateur « \leftarrow » représente la mise à jour

$$\text{Où : } \delta \cdot W_i = \frac{\partial E}{\partial W_i} = \delta \cdot W_i$$

représente le taux ou le pas d'apprentissage, il a une faible magnitude et est compris entre [0 et 1].

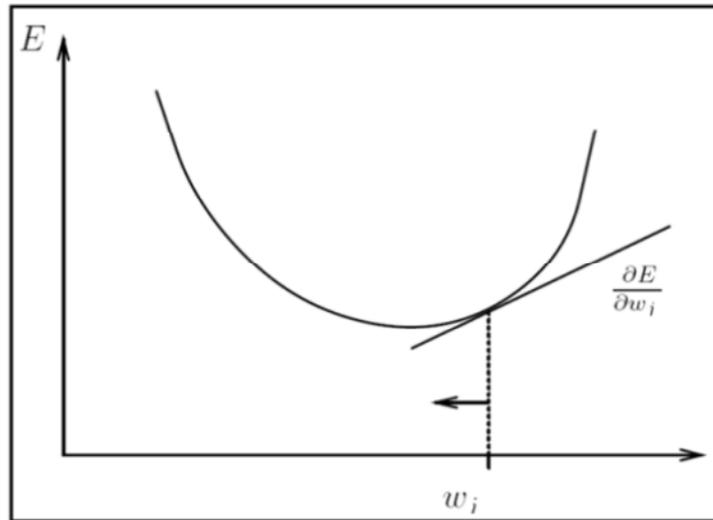


Figure (1-5) Gradient de l'erreur totale.

5. Continuer les étapes 2 et 3 tant que l'écart entre les W_i de deux époques successives est grand. Une époque représente un cycle propagation - rétro-propagation, appelée aussi itération.

Calculons à présent ΔW_i .

On suppose que la couche de sortie ne comporte qu'un neurone. L'erreur quadratique moyenne E vaut:

$$\begin{aligned} \frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \cdot \frac{1}{2} \cdot \delta^2 = \delta \cdot \frac{\partial \delta}{\partial w_i} \\ &= \delta \cdot \frac{\partial}{\partial w_i} (d - y(h)) \\ &= \delta \cdot \left(- \frac{\partial y(h)}{\partial w_i} \right) \\ &= \delta \cdot \left(- \frac{\partial f(\sum_{i=1}^n W_i \cdot x_j)}{\partial w_i} \right) \end{aligned}$$

$f\left(\sum_{i=1}^n W_i \cdot x_j\right)$ Est une fonction composée. On note que

$$(x) = \sum_{i=1}^n W_i \cdot x_j.$$

$f\left(\sum_{i=1}^n W_i \cdot x_j\right)$ Peut être écrite sous la forme $(f \circ \sigma)(x)$, sa dérivée est donc

$$\sigma'(x) \cdot f'(\sigma(x))$$

De ce fait on peut écrire :

$$\frac{\partial E}{\partial W_i} = -\delta \cdot f'\left(\sum_{i=1}^n W_i \cdot x_j\right) \cdot x_i$$

La mise à jour des poids synaptiques se fait systématiquement :

$$W_i \rightarrow W_i + \alpha \cdot \delta \cdot f'\left(\sum_{i=1}^n W_i \cdot x_j\right) \cdot x_i$$

Si la fonction d'activation est de type logistique-sigmoïde (Tableau. 1.1) le calcul de sa dérivée est très simple :

$$\left(\frac{1}{1+e^{-\left(\sum_{i=1}^n W_i \cdot x_j\right)}}\right)' = f\left(\sum_{i=1}^n W_i \cdot x_j\right)[1 - f\left(\sum_{i=1}^n W_i \cdot x_j\right)]$$

Remplaçons I.6 dans I.5, on obtient :

$$\frac{\partial E}{\partial W_i} = -\delta \cdot f\left(\sum_{i=1}^n W_i \cdot x_j\right)[1 - f\left(\sum_{i=1}^n W_i \cdot x_j\right)] \cdot x_i$$

La deuxième fonction d'activation qui est largement utilisée, est la fonction tanh-sigmoïde (Tableau. 1.1). Sa dérivée est aussi relativement facile à calculer

$$\frac{d(\tanh(\sum_{i=1}^n W_i \cdot x_j))}{d(\sum_{i=1}^n W_i \cdot x_j)} = \frac{4}{(e^{\sum_{i=1}^n W_i \cdot x_j} + e^{-\sum_{i=1}^n W_i \cdot x_j})^2}$$

Remplaçons I.7 dans I.6, on obtient :

$$\Delta W_i = \alpha \cdot \delta \cdot \frac{4}{(e^{\sum_{i=1}^n W_i \cdot x_j} + e^{-\sum_{i=1}^n W_i \cdot x_j})^2} \cdot x_i$$

1.3.3.1.2. Cas du Perceptron multicouche

Le PMC a une structure bien particulière: ses neurones sont organisés en couches successives (figure. 1.6.). Chaque neurone d'une couche reçoit des signaux de la couche précédente et transmet le résultat à la suivante, si elle existe. Les neurones d'une même couche ne sont pas interconnectés. Un neurone ne peut donc envoyer son résultat qu'à un neurone situé dans une couche postérieure à la sienne.

L'orientation du réseau est fixée par le sens unique de propagation de l'information, de la couche d'entrée vers la couche de sortie. Pour les réseaux considérés, les notions de couches d'entrée et de sortie sont donc systématiques. Dans ce qui suit un RNA à une couche cachée est utilisé (figure. 1.6).

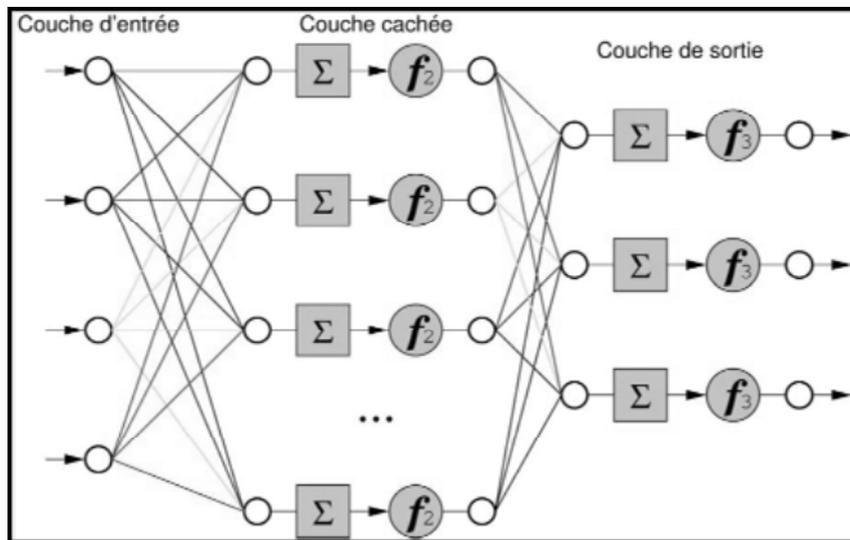


Figure (1-6) Perceptron multicouche

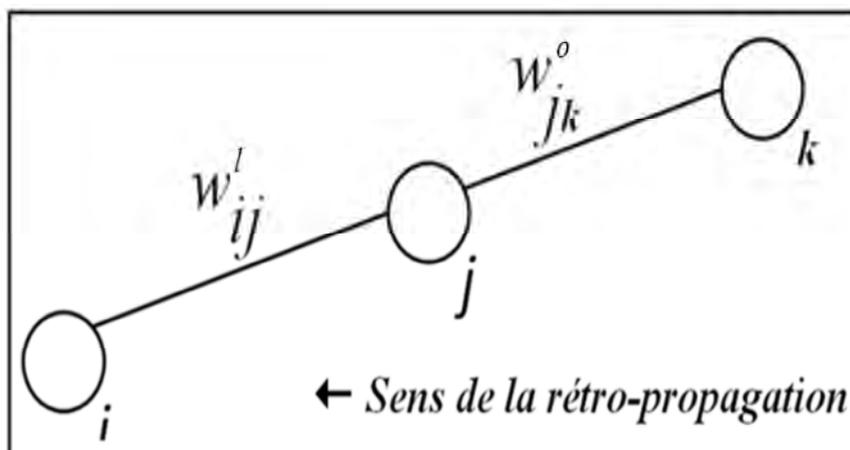


Figure (1-7) Connexions entre les neurones d'un PMC

i. Correction de l'erreur pour la couche de sortie

Soit l'erreur à la sortie du neurone k :

$$e_k = d_k - z_k$$

Avec :

d_k : la sortie désirée

z_k : la sortie simulée par le RNA

L'objectif ici est d'adapter les poids synaptiques de manière à minimiser la somme des erreurs sur tous les neurones de sortie. La somme de l'erreur quadratique moyenne sur l'ensemble c des neurones de sortie et est définie par :

$$E = \frac{1}{2} \sum_{k=1}^c e_k^2$$

La sortie du neurone j est

$$z_k = f_3(h_k) = f_3[\sum_{j=1}^r W_{jk}^o \cdot y_j]$$

- $f_3(h_k)$ est la fonction d'activation au niveau de la couche de sortie.

- h_k est la somme pondérée des entrées du neurone k .

- W_{jk}^o est le poids de la connexion entre le neurone j de la couche cachée et le neurone k de la couche de sortie. o est l'indice de la couche de sortie.

- r : nombre de neurones de la couche cachée.

- c : nombre de neurones de la couche de sortie.

Comme on a fait avec le perceptron, la correction de l'erreur est effectuée en modifiant le

poids W_{jk}^o dans le sens opposé au gradient $\frac{\partial E}{\partial W_{jk}^o}$ de l'erreur.

La dérivée partielle de la fonction composée E se fait cette fois par la règle de chaînage, soit :

$$\frac{\partial E}{\partial W_{jk}^o} = \frac{\partial E}{\partial e_k} \cdot \frac{\partial e_k}{\partial z_k} \cdot \frac{\partial z_k}{\partial h_k} \cdot \frac{\partial h_k}{\partial W_{jk}^o}$$

On garde la même notation de la variation de poids

$$\Delta W_{jk}^o = -\alpha_2 \frac{\partial E}{\partial W_{jk}^o}$$

Calculons à présent les termes du gradient :

$$\frac{\partial E}{\partial e_k} = \frac{\partial [\frac{1}{2} \sum_{k \in c} e_k^2]}{\partial e_k} = \frac{1}{2} \frac{\partial e_k^2}{\partial e_k} = e_k$$

$$\frac{\partial e_k}{\partial z_k} = \frac{\partial (d_z - z_k)}{\partial z_k} = -1$$

On prend à présent une fonction d'activation de type logistique sigmoïde qui a la forme :

$$f_3(h_k) = \frac{1}{1+e^{-h_k}}$$

$$\begin{aligned} \frac{\partial z_k}{\partial h_k} &= \frac{\partial \left[\frac{1}{1+e^{-h_k}} \right]}{\partial h_k} \\ &= \frac{e^{-h_k}}{(1+e^{-h_k})^2} \end{aligned}$$

$$= z_k \cdot \frac{e^{-h_k}}{(1+e^{-h_k})}$$

$$\frac{\partial z_k}{\partial h_k} = z_k \cdot (1 - z_k)$$

$$= z_k \cdot \left[\frac{e^{-h_k+1}}{(1+e^{-h_k})} - \frac{1}{(1+e^{-h_k})} \right]$$

Finalement

$$\frac{\partial h_k}{\partial W_{jk}^o} = \frac{\partial [\sum_{j=1}^r W_{jk}^o \cdot y_j]}{\partial W_{jk}^o} = y_j$$

On obtient donc :

$$\frac{\partial E}{\partial W_{jk}^o} = -e_k \cdot [z_k \cdot (1 - z_k)] \cdot y_j$$

Et la règle dite de « DELTA » pour la couche de sortie s'exprime donc par :

$$\Delta W_{jk}^o = -\alpha_2 \frac{\partial E}{\partial W_{jk}^o} = -\alpha_2 \cdot \delta_k \cdot y_j$$

Avec :

$$\delta_k = e_k \cdot z_k \cdot [1 - z_k]$$

δ_k S'appelle le gradient local et α_2 représente le pas d'apprentissage de la couche de sortie. Ce résultat a la même forme que celui obtenu par le Perceptron.

ii. Correction de l'erreur pour la couche cachée

Considérons désormais le cas des neurones sur la dernière couche cachée (le cas des autres couches cachées est semblable).

L'expression de la dérivée partielle de l'erreur totale E par rapport à w_{ij}^l reste la même que précédemment mais dans ce cas on ne dérive pas par rapport à e_i car celle-ci est inconnue :

$$\frac{\partial E}{\partial w_{ij}^l} = \frac{\partial E}{\partial y_j} \cdot \frac{\partial y_j}{\partial m_j} \cdot \frac{\partial m_j}{\partial w_{ij}^l}$$

- m_j est la somme pondérée des entrées du neurone j qui est égale à :

$$m_j = \sum_{i=1}^n w_{ij}^l \cdot x_i$$

- w_{ij}^l est le poids de la connexion entre le neurone i de la couche d'entrée et le neurone j de la couche cachée. l désigne symboliquement la couche cachée.

L'indice k représentera un neurone sur la couche de sortie comme le montre la figure. 1.7.

Le 2^{ème} et le 3^{ème} terme de l'équation 1-20 sont équivalents à 1-15 et 1-16 respectivement, tout en changeant les indices qui correspondent à la couche cachée :

$$\frac{\partial y_j}{\partial m_j} = y_j [1 - y_j]$$

$$\frac{\partial m_j}{\partial w_{ij}^l} = x_i$$

Pour le 1^{ère} terme, la procédure suivante est utilisée:

$$\frac{\partial E}{\partial y_j} = \frac{\partial [\frac{1}{2} \sum_{k=1}^c e_k^2]}{\partial y_j}$$

c : représente toujours le nombre de neurones de la couche de sortie.

Le problème ici, contrairement au cas des neurones de la couche de sortie, est e_i dépend de y_j . On ne peut donc pas se débarrasser de cette somme.

Par contre on peut écrire :

$$\begin{aligned} \frac{\partial E}{\partial y_j} &= \sum_{k \in c} [e_k \cdot \frac{\partial e_k}{\partial y_j}] \\ &= \sum_{k \in c} [e_k \cdot \frac{\partial e_k}{\partial h_k} \cdot \frac{\partial h_k}{\partial y_j}] \\ &= \sum_{k \in c} [e_k \cdot \frac{\partial [d_k - z_k]}{\partial h_k} \cdot \frac{\partial \sum_{j=1}^r (W_{jk}^o \cdot y_j)}{\partial y_j}] \end{aligned}$$

On prend comme fonction d'activation f_2 la fonction logistique sigmoïde :

$$= \sum_{k \in c} [e_k \cdot (-z_k) \cdot (1 - z_k) \cdot W_{jk}^o]$$

Et en substituant l'équation (1.19) on obtient :

$$\frac{\partial E}{\partial y_j} = - \sum_{k=1}^c \delta_k \cdot W_{jk}^o$$

En remplaçant l'équation (I.21) dans (I.20) on trouve

$$\frac{\partial E}{\partial W_{ij}^l} = -y_j \cdot [1 - y_j] \cdot [\sum_{k=1}^c \delta_k \cdot W_{jk}^o] \cdot x_i$$

Et

$$\Delta W_{ij}^l = -\alpha_1 \cdot \frac{\partial E}{\partial W_{ij}^l} = -\alpha_1 \cdot \delta_j \cdot x_i$$

Avec :

$$\delta_j = y_j \cdot [1 - y_j] \cdot [\sum_{k=1}^c \delta_k \cdot W_{jk}^o]$$

δ_j et α_1 représentent le gradient local et le pas d'apprentissage respectivement de la couche cachée.

Les deux expressions restent valables pour toutes les couches cachées.

2.3.3.1.3 Sommaire de l'algorithme des DELTA

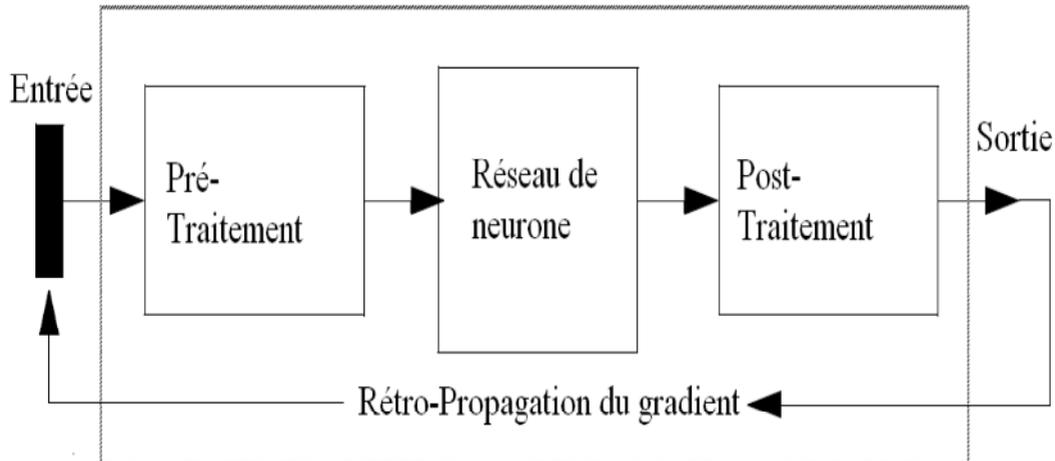


Figure (1-8) Procédure d'élaboration d'un modèle neuronal

La procédure de l'élaboration de l'algorithme de la rétro-propagation du gradient dans sa version de base se résume comme suit :

- Initialisation des poids à de petites valeurs aléatoires dans l'intervalle $[-0.5, 0.5]$
- Normalisation des données d'apprentissage pour augmenter les performances.
Cette étape est fortement recommandée pour éviter la saturation des fonctions d'activation utilisées, qui sont généralement dans le perceptron multicouche de type sigmoïde.
- Pour chaque donnée d'apprentissage q :
 - Calculer les sorties simulées en propageant les variables de l'entrée vers la sortie
 - Ajuster les poids en rétro-propageant l'erreur calculée :

$$W_{jk}^o(q+1) = W_{jk}^o(q) - \quad W_{jk}^o(q) = W_{jk}^o(q) + \alpha_2 \cdot \delta_k(q) \cdot y_j(q) \quad \text{Pour la couche de sortie}$$

$$W_{ij}^l(q+1) = W_{ij}^l(q) - \quad W_{ij}^l(q) = W_{ij}^l(q) + \alpha_1 \cdot \delta_j(q) \cdot x_i(q) \quad \text{Pour la couche cachée}$$

Répéter cette étapes jusqu'à un nombre maximum d'itérations (époques) ou jusqu'à ce que la racine de l'erreur quadratique moyenne (MSE) soit inférieure à un certain seuil fixé par le concepteur du RNA.

1.3.3.1.4 Règle de « DELTA généralisé »

La convergence du réseau par rétro-propagation est un problème crucial car il requiert de nombreuses itérations. Un paramètre est souvent ajouté pour accélérer la convergence. Ce paramètre est appelé « le momentum ».

Les deux équations qui décrivent la mise à jour des poids synaptiques dans la couche de sortie et la couche cachée avec la règle de « DELTA généralisé » sont définis par :

$$W_{jk}^o(q+1) = W_{jk}^o(q) - \Delta W_{jk}^o(q) \quad W_{jk}^o(q) = W_{jk}^o(q) + \alpha_2 \cdot \delta_k(q) \cdot y_j(q) + \beta_2 \cdot W_{jk}^o(q)$$

$$W_{ij}^l(q+1) = W_{ij}^l(q) - \Delta W_{ij}^l(q) \quad W_{ij}^l(q) = W_{ij}^l(q) + \alpha_1 \cdot \delta_j(q) \cdot x_i(q) + \beta_1 \cdot W_{ij}^l(q)$$

Où β_1 et β_2 sont compris entre [0 et 1] et sont les momentums qui représentent une espèce d'inertie lors du chargement de poids.

L'algorithme de la rétro-propagation modifie les poids à partir de $\partial E / \partial W$. En fait, ce dernier terme permet d'accélérer la convergence (variation plus grande du changement du poids) lorsqu'on est loin du minimum. En pratique, la méthode du gradient DELTA généralisé peut être efficace lorsque l'on est loin du minimum. Quand on s'en approche, la norme du gradient diminue et donc l'algorithme progresse plus lentement.

Pour optimiser la vitesse de convergence, plusieurs règles ont été proposées telles que :

1.3.3.1.5. Techniques de réglage du pas

a) Technique du pas constant : elle consiste à adopter un pas constant $\beta = \beta_{cte}$ tout au long de l'algorithme. Elle est très simple mais peu efficace puisqu'elle ne prend pas en considération la décroissance de la norme du gradient.

b) Technique du pas asservi : on peut asservir le pas à l'aide de la norme du gradient de sorte que le pas évolue en sens inverse de celle-ci. A chaque étape, le pas peut être calculé par

$$\beta = \frac{\beta_{cte}}{1 + \|\mathbf{E}\|}$$

1.3.3.1.6. Résilient back propagation

Habituellement, nous utilisons une fonction d'activation sigmoïde pour la couche cachée. Cette fonction est caractérisée par le fait que la pente approche la valeur zéro lorsque les données en entrée ont des valeurs très élevées. Si le gradient a une valeur très petite, cela

occasionne de très petits changements dans les poids et les biais. Pour palier à ce problème, seul le signe de la dérivée est pris en compte et non la valeur de la dérivée.

1.3.3.1.7. Méthode d'apprentissage par algorithme du gradient conjugué

Cette méthode initiée par Hestenes and Stiefel en 1952 [7], est une méthode itérative qui se base sur la recherche de directions successives permettant d'atteindre la solution exacte d'un système de matrice symétrique et définie positif, elle est intéressante pour les RNA à un grand nombre de neurones : convergence rapide et est modeste en mémoire.

1.3.3.1.8 Les méthodes de gradient du second ordre

Les méthodes que nous venons de décrire sont simples mais en général peu efficaces.

Nous avons donc systématiquement recours à l'utilisation de méthodes plus performantes (pour une comparaison numérique entre ces méthodes, voir [8]). Elles sont dites du second ordre parce qu'elles prennent en considération la dérivée seconde de la fonction de coût.

Algorithmes de BFGS et de Levenberg-Marquardt

L'algorithme de BFGS (du nom de ses inventeurs : Broyden, Fletcher, Goldfarb et Shanno) fait partie des méthodes d'optimisation dites "quasi-newtoniennes". Ces méthodes sont une généralisation de la méthode de Newton.

La méthode de Newton consiste à l'application de la règle suivante :

$$W(q + 1) = W(q) + [E^2(W(q))]^{-1} \cdot E(W(q))$$

Le terme $[E^2(W(q))]^{-1}$ représente l'inversion de la matrice Hessienne (la dérivée seconde) de la fonction E calculée avec le vecteur des paramètres disponibles à l'époque courante. La direction de descente est dans ce cas :

$$d_{q+1} = [E^2(W(q))]^{-1} \cdot E(W(q))$$

Néanmoins, cette méthode de Newton représente un inconvénient, elle ne peut pas converger en une seule itération. De plus, cette méthode nécessite l'inversion de la matrice Hessienne à chaque époque, ce qui conduit à des calculs lourds [9] et à une instabilité numérique. On utilise de préférence une méthode de "quasi-Newton".

Les méthodes de quasi-Newton consistent à approcher l'inverse du hessien plutôt que de calculer sa valeur exacte.

L'algorithme de Levenberg-Marquardt (LM), est une méthode « quasi-newtonienne ». C'est une règle d'ajustement des paramètres qui a l'expression suivante :

$$W(q + 1) = W(q) + \mu_{q+1} \cdot M_{q+1} \cdot \nabla E(W(q))$$

μ_{q+1} est une approximation, calculée itérativement, de l'inverse de la matrice Hessienne.

Où :

$$M_{q+1} = M_q + \left[1 + \left(\frac{\gamma_q^T \cdot M_q \cdot \gamma_q}{\delta_q^T \cdot \gamma_q} \right) \right] \cdot \frac{\delta_q^T \cdot \delta_q}{\delta_q^T \cdot \gamma_q} + \frac{\delta_q \cdot \gamma_q^T \cdot M_q + M_q \cdot \gamma_q \cdot \delta_q^T}{\delta_q^T \cdot \gamma_q}$$

$$\nabla E(W(q + 1)) = \nabla E(W(q)) + \nabla^2 E(W(q)) \delta_{q+1}$$

$$\delta_{q+1} = W(q + 1) - W(q)$$

La valeur initiale de M_{q+1} est la matrice identité. Si, à une itération, la matrice calculée n'est pas définie positive, elle est réinitialisée à la matrice identité.

Le LM est un standard pour l'optimisation de l'erreur quadratique due à ses propriétés de convergence rapide et de robustesse. Cette méthode s'appuie sur les techniques des moindres carrés non-linéaires et de l'algorithme de GAUSS-NEWTON à voisinage restreint. En fait, la méthode LM est un condensé de deux techniques exposées précédemment. En effet, cette méthode tend vers la méthode de Newton pour une valeur de μ_{q+1} petite mais est équivalente à la méthode du gradient « DELTA généralisé » pour un pas $\mu_{q+1} = \frac{1}{\mu_{q+1}}$ pour une valeur de

μ_{q+1} grande. Le Hessien est toujours défini positif ce qui assure la convergence vers un minimum de la solution. Par contre, le volume de calculs nécessaires à chaque itération de cet algorithme croît rapidement avec le nombre de paramètres. Pour plus de détails sur cet algorithme consulter [3].

Pour ce qui est du choix décisif entre le BFGS et LM, il s'est avéré nécessaire de faire une comparaison entre les valeurs générées par ces deux modèles et les mesures. Les outils de comparaison sont les paramètres statistiques représentés dans la section suivante.

1.4 Conclusion

La technique des RNA est une puissante méthode de régression non linéaire. Le tableau suivant résume les équivalences de vocabulaire entre les méthodes de réseaux de neurones et statistiques.

Tableau (1-2) Glossaire réseaux de neurones / statistiques [10] .

Réseau de neurones	Statistiques
Architecture	Fonction de régression
Apprentissage	Estimation
Poids	Paramètres
Apprentissage supervisé	Régression / Classification
Réseau de neurones	Modèle
Grand : 100 000 poids	Grand : 50 paramètres
Ensemble d'apprentissage	Observations / Echantillon
Grand : 50 000 exemples	Grand : 200 cas
Généralisation	Interpolation, extrapolation

Les possibilités, la commodité et la précision offertes par les RNA sont utilisées avec succès en génie civil dans les domaines des structures, de la géotechnique et des bétons.

La Figure (1-9) montre le nombre de publications dans le Journal of Computing in Civil Engineering qui comportent dans leur titre le mot réseau de neurones.

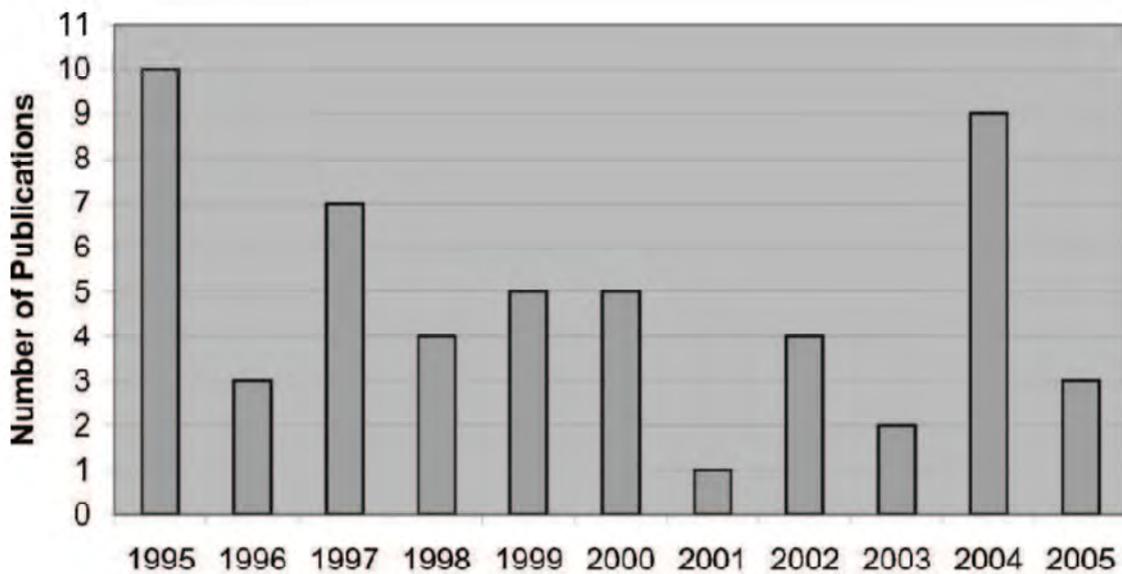


Figure (1-9) Nombre d'article publiés qui utilisent dans leur titre le terme (neurones) dans le journal of computing in civil engineering [11] .

2. HISTORIQUE ET APPLICATIONS DES RESEAUX DE NEURONES

2.1. Historique

L'origine de l'inspiration des réseaux de neurones artificiels remonte à 1890 où W. James[18], célèbre psychologue américain, introduit le concept de mémoire associative. Il propose ce qui deviendra une loi de fonctionnement pour l'apprentissage des réseaux de neurones, connue plus tard sous le nom de loi de Hebb. Le champ des réseaux neuronaux va démarrer par la présentation en 1943 par W. McCulloch et W. Pitts du neurone formel [12].

➤ Le neurone formel de McCulloch et Pitts

Le premier modèle mathématique et informatique du neurone biologique est proposé par Warren McCulloch et Walter Pitts en 1943. En s'appuyant sur les propriétés des neurones biologiques connues à cette époque, issues d'observations neurophysiologiques et anatomiques, McCulloch et Pitts proposent un modèle simple de neurone formel. Il s'agit d'un neurone binaire, c'est-à-dire dont la sortie vaut 0 ou 1. Pour calculer cette sortie, le neurone effectue une somme pondérée de ses entrées (qui, en tant que sorties d'autres neurones formels, valent aussi 0 ou 1) puis applique une fonction d'activation à seuil : si la somme pondérée dépasse une certaine valeur, la sortie du neurone est 1, sinon elle vaut 0.

McCulloch et Pitts étudiaient en fait l'analogie entre le cerveau humain et les machines informatiques universelles. Ils montrèrent en particulier qu'un réseau (bouclé) constitué des neurones formels de leur invention a la même puissance de calcul qu'une machine de Turing.

Malgré la simplicité de cette modélisation, ou peut-être grâce à elle, le neurone formel dit de McCulloch et Pitts reste aujourd'hui un élément de base des réseaux de neurones artificiels. De nombreuses variantes ont été proposées, plus ou moins biologiquement plausibles, mais s'appuyant généralement sur les concepts inventés par les deux auteurs. On sait néanmoins aujourd'hui que ce modèle n'est qu'une approximation des fonctions remplies par le neurone réel et, qu'en aucune façon, il ne peut servir pour une compréhension profonde du système nerveux.

➤ Formulation mathématique

On considère le cas général d'un neurone formel à m entrées, auquel on doit donc soumettre les m grandeurs numériques (ou signaux, ou encore stimuli) notées x_1 à x_m . Un modèle de

neurone formel est une règle de calcul qui permet d'associer aux m entrées une sortie : c'est donc une fonction à x_m variables et à valeurs réelles.

Dans le modèle de McCulloch et Pitts, à chaque entrée est associé un poids synaptique, c'est-à-dire une valeur numérique notée de \mathcal{W}_1 pour l'entrée 1 jusqu'à \mathcal{W}_m pour l'entrée m . La première opération réalisée par le neurone formel consiste en une somme des grandeurs reçues en entrées, pondérées par les coefficients synaptiques, c'est-à-dire la somme

$$\mathcal{W}_1 x_1 + \dots + \mathcal{W}_m x_m = \sum_{j=1}^m \mathcal{W}_j x_j .$$

À cette grandeur s'ajoute un seuil \mathcal{W}_0 . Le résultat est alors transformé par une fonction d'activation non linéaire (parfois appelée fonction de sortie), φ La sortie associée aux entrées x_1 à x_m . Est ainsi donnée par

$$\varphi \left(\mathcal{W}_0 + \sum_{j=1}^m \mathcal{W}_j x_j \right),$$

Qu'on peut écrire plus simplement :

$$\varphi \left(\sum_{j=0}^m \mathcal{W}_j x_j \right) ,$$

En ajoutant au neurone une entrée fictive x_0 fixée à la valeur 1.

Dans la formulation d'origine de McCulloch et Pitts, la fonction d'activation est la fonction de Heaviside (fonction en marche d'escalier), dont la valeur est 0 ou 1. Dans ce cas, on préfère parfois définir la sortie par la formule suivante

$$\varphi \left(\sum_{j=1}^m \mathcal{W}_j x_j - \mathcal{W}_0 \right) ,$$

qui justifie le nom de seuil donné à la valeur \mathcal{W}_0 . En effet, si la somme $\sum_{j=1}^m \mathcal{W}_j x_j$ dépasse \mathcal{W}_0 la sortie du neurone est 1, alors qu'elle vaut 0 dans le cas contraire : \mathcal{W}_0 est donc le seuil d'activation du neurone, si on considère que la sortie 0 correspond à un neurone « éteint ».

➤ 1949 : D. Hebb, physiologiste américain explique le conditionnement chez l'animal par les propriétés des neurones eux-mêmes. Ainsi, un conditionnement de type pavlovien tel que, nourrir tous les jours à la même heure un chien, entraîne chez cet animal la sécrétion de salive à cette heure précise même en l'absence de nourriture. La loi de modification des propriétés des connexions entre neurones qu'il propose explique en partie ce type de résultats expérimentaux [13] .

2.1.1. Les premiers succès

➤ 1957 : F. Rosenblatt développe le modèle du Perceptron . Il construit le premier neuro ordinateur basé sur ce modèle et l'applique au domaine de la reconnaissance de formes. notons qu'à cet époque les moyens à sa disposition sont limités et c'est une prouesse technologique que de réussir à faire fonctionner correctement cette machine plus de quelques minutes , le perceptron se voulait un modèle de l'activité perspective[6]. Il se compose d'une rétine et d'une couche qui donne la réponse correspondant à la simulation donnée en entrée.

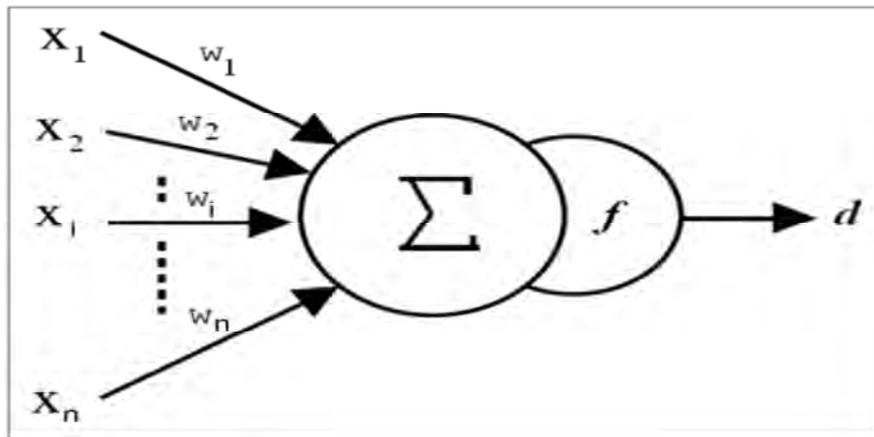


Figure (2-1) : modèle de l'activité perspective

ADALINE (Adaptive Linear Neuron ou plus tard Adaptive Linear Element) est un réseau de neurones artificiels simple-couche.

Adaline s'appuie sur le neurone formel de McCulloch et Pitts. Il se compose d'un poids synaptique, d'un biais (une constante qu'on ajoute à l'entrée) et d'une fonction de sommation.

La différence entre Adaline et le modèle de McCulloch-Pitts réside dans la phase d'apprentissage, où les poids synaptiques sont ajustés en fonction de la somme pondérée des données en entrées. Dans le perceptron standard (McCulloch-Pitts), le réseau passe par la fonction d'activation (ou fonction de transfert) et la fonction de sortie est utilisée pour ajuster les poids synaptiques, Il existe aussi une extension appelée Madaline.

Adaline est un réseau neuronal simple-couche, avec plusieurs nœuds, chacun acceptant plusieurs entrées et générant une sortie. Étant données les variables suivantes :

x , le vecteur d'entrée

w , le « vecteur poids »

n , le nombre d'entrées

θ , une constante

y , la sortie

on a alors : $y = \sum_{j=1}^n x_j W_j + \theta$.

Algorithme d'apprentissage

Posons :

η , le taux d'apprentissage (une constante)

d , la sortie désirée

o , la sortie réelle,

alors les poids sont mis à jour comme suit : $w \leftarrow w + \eta (d - o) x$.

Adaline converge vers l'erreur des moindres carrés qui est : $E = (d - o)^2$

➤ En 1969, dans *Perceptrons*, Marvin Minsky coécrit avec Seymour Papert pour critiquer Frank Rosenblatt, il a montré les limites des réseaux de neurones de type perceptron, notamment l'impossibilité de traiter des problèmes non linéaires ou de connectivité. Ceci a eu pour conséquence de drainer l'essentiel des crédits de recherche vers l'intelligence artificielle symbolique[14].

Marvin Minsky défend l'idée que l'intelligence artificielle doit utiliser des approches multiples, notamment pour la représentation des connaissances, au lieu de se limiter à une seule approche qui serait censée être la meilleure. Les systèmes doivent disposer de "gestionnaires", capable de sélectionner les meilleures solutions à un problème donné.

Pour représenter les connaissances, Marvin Minsky a développé le concept de frame et le Frame representation language.

Dans *La Société de l'Esprit*, Marvin Minsky présente l'esprit comme une architecture d'agents élémentaires, indépendants, mais surtout hiérarchisés. Les agents les plus courants sont les lignes (K-lines), agents de mémoire à court terme servant à activer un ensemble d'agents donné. L'activation de lignes K peut par exemple permettre de retrouver une configuration particulière d'agents qui s'est avérée efficace.

Les nèmes sont des agents représentant les connaissances, alors que les nommes sont des agents capables de traiter les connaissances. Les polynèmes permettent d'activer des agents

représentant des aspects différents d'un même objet. Les paranomes permettent de manipuler simultanément différents modes de représentations des connaissances.

Ces agents de base vont se combiner pour former des agencements de plus grande taille capables d'opérations complexes (frames, frame-arrays, transframes). Minsky introduit la notion de "cerveau B", dont le rôle est de surveiller et d'examiner non pas le monde extérieur, mais l'esprit lui-même (cerveau A), en corrigeant les erreurs ou en mettant fin à une activité mentale improductive (boucles, répétitions). Cette division de l'esprit en "niveaux de réflexion" fait l'objet de nouveaux développements dans les travaux plus récents de Minsky.

Un système intelligent disposera de deux types d'agents, des spécialistes et des gestionnaires. Les spécialistes peuvent implémenter les techniques connues d'intelligence artificielle pour résoudre des problèmes précis (systèmes à base de règles, réseaux sémantiques, réseaux bayésiens). Les gestionnaires sont chargés de planifier, de sélectionner les spécialistes, d'évaluer les résultats obtenus.

Leur objectif est atteint, il y a abandon financier des recherches dans le domaine (surtout aux U.S.A.), les chercheurs se tournent principalement vers l'IA et les systèmes à bases de règles.

2.1.2. L'ombre

➤ 1967-1982 : Toutes les recherches ne sont, bien sûr, pas interrompues. Elles se poursuivent, mais déguisées, sous le couvert de divers domaines comme : le traitement adaptatif du signal, la reconnaissance de formes, la modélisation en neurobiologie, etc.

De grands noms travaillent durant cette période telle : S. Grossberg, T. Kohonen, ... dont nous reparlerons.

➤ Cartes de Kohonen & loi de Grossberg

Les cartes topologiques de Kohonen correspondent à un apprentissage non-supervisé.

Les cartes topologiques de Kohonen sont des cartes auto-adaptatives dont la configuration spatiale possède une importance dans leur mode de fonctionnement. Ce modèle utilise la loi de Grossberg, qui est une version modifiée de la loi de Hebb prenant en compte leur configuration citée ci-dessus. Les contraintes géométriques permettent de favoriser certaines cellules de cette carte et d'en inhiber d'autres.

Les cartes de Kohonen ont la même construction qu'un perceptron : 1 couche d'input où chaque cellule est reliée à toutes les cellules de la 2^{de} couche, la couche d'output. Cependant, la couche d'output de ce modèle est dite « compétitive ». C'est là que va se

dérouler l'essentiel du travail du réseau et c'est la géométrie de cette couche qui fait la spécificité de ce modèle. Les cartes peuvent avoir une forme hexagonale ou rectangulaire.

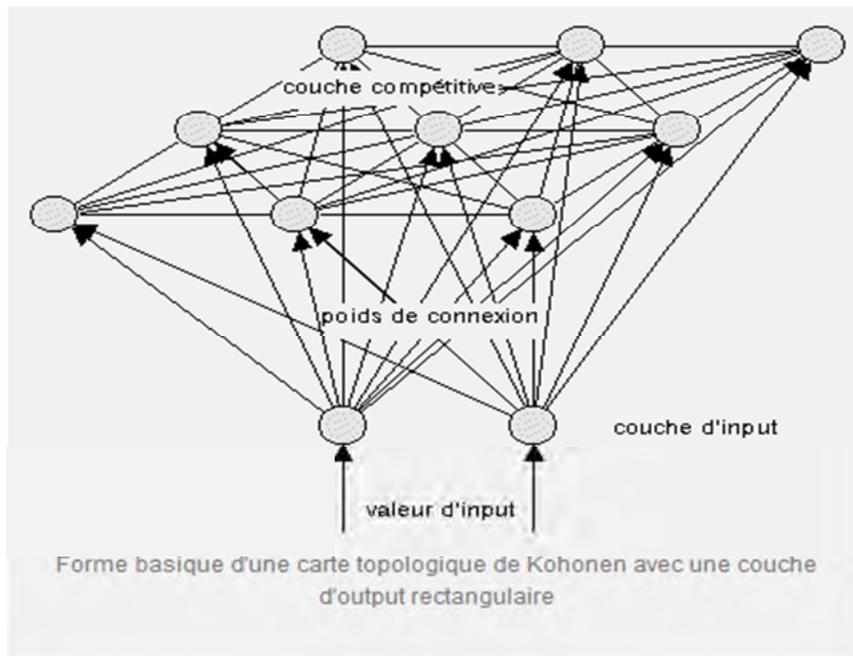


Figure (2-2) : forme basique d'une carte topologique de Kohonen

Chaque cellule de la couche compétitive se regroupe selon leur similitudes. Ce phénomène peut se schématiser de la façon suivante :

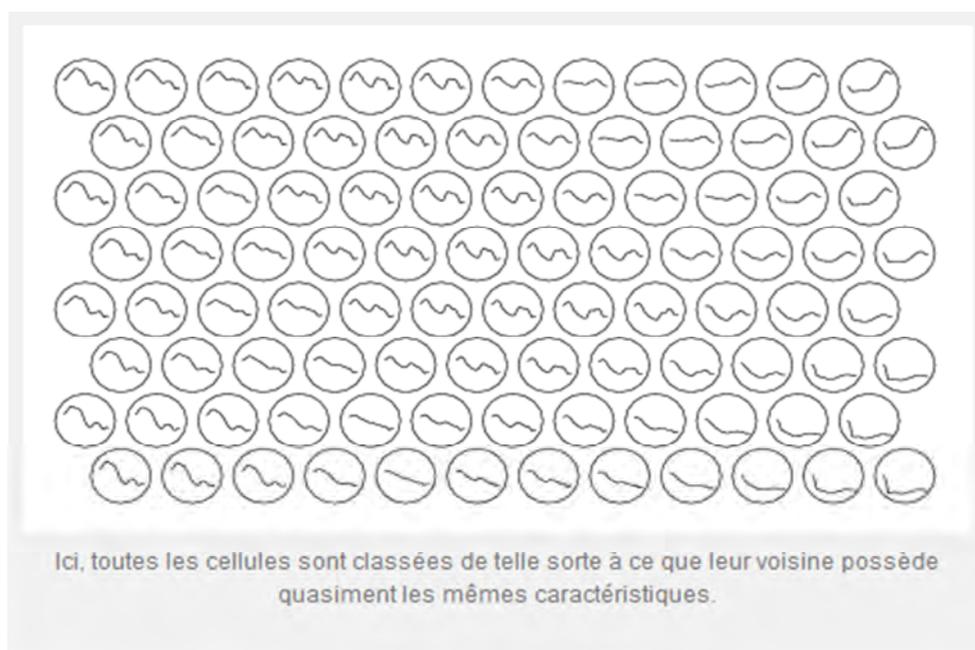


Figure (2-3) : organisation des cellules de modèle de Kohonen

Ce type d'organisation se retrouve dans le cerveau, comme dans la région du cortex visuel où deux zones proches dans le cerveaux correspondent à deux zones proches de la rétine, ce qui a été démontré en 1947 par David Hubel et Torsten Wiesel. On peut donc constater ici que chaque neurone se spécialise : il ne répondra pas à toutes les stimulations que provoqueront les input proposées, seulement à certaines.

La méthode d'apprentissage de ce réseau est dite « compétitive » puisqu'à chaque vecteur d'entrée qui lui est présenté dans le processus d'apprentissage est calculé, en fonction du vecteur présenté, un « gagnant » qui pourra modifier le poids de ses connexions avec les cellules environnantes. Ce gagnant est élu selon la formule suivante : où W_r' et W_r représentent le poids des connexions au sein du réseau et v le vecteur d'input.

Les cellules adjacentes au gagnant dans un certain périmètre pourront elles aussi modifier le poids de leur connexion. Plus la cellule est loin, plus la modification apportée à ses connexions est faible. La modification de ces poids est définie par la formule suivante :

$\Delta w_{ij} = R(a_j - w_{ij})a_i$ où W_{ij} est le poids de la connexion entre les neurones i et j , et a_j et a_i sont les valeurs respectives d'activation des neurones j et i et R est une constante positive appelée « force d'apprentissage ».

Il existe par ailleurs un mécanisme pénalisant les cellules trop souvent désignées gagnantes.

Dans son fonctionnement, le modèle de Hopfield reflète bien la plasticité du cerveau puisqu'à chaque problème qui lui sera soumis, il proposera une réponse en organisant ses cellules de différentes manières. L'output proposée est alors une projection en 2D ou en 3D de ces cellules[17].

Grossberg cherchait à comprendre de quelle façon les sensations deviennent des perceptions, en passant de la mémoire à court terme à la mémoire à long terme. Les cartes de Kohonen découlent d'observations purement biologiques à propos du système de perception des mammifères : leurs travaux se rejoignent donc.

L'organisation par régions de la rétine et du cortex visuel (les régions se correspondant) dont nous parlions plus haut se retrouvent pour l'ouïe ou encore le goût et l'odorat. Néanmoins, cette organisation n'est pas génétique : elle n'est pas présente chez l'humain basiquement mais découle d'un apprentissage des personnes.

Les cartes de Kohonen fonctionnent de la même manière. Le premier « apprentissage » du réseau concerne d'abord sa structure, car il va devoir organiser ses cellules de la bonne

manière selon la règle évoquée plus haut. Ensuite, et ensuite seulement, l'apprentissage plus classique va pouvoir se faire, au moyen de la règle d'apprentissage de Hebb modifiée. Mais cet apprentissage ne pourrait avoir lieu sans le premier au vu de l'importance de la place de chaque cellule au sein des cartes topologiques. Cela agit de la même façon que chez l'humain : tout d'abord, en prenant le cortex visuel pour exemple, les zones vont devoir se mettre en place en fonction des zones de la rétine. Ce n'est qu'ensuite que l'on apprend à discerner les formes et les couleurs, et qu'enfin, à un stade plus avancé (après une année environ et grâce à la parole), on apprend le vocabulaire correspondant ; mais nous ne sommes alors plus dans le même modèle d'apprentissage. C'est ce qui explique que, durant les premiers mois de sa vie, un bébé y voie mal.

2.1.3. Les renouvelaux

➤ Loi de Hebb & modèle de Hopfield

La loi de Hebb et le modèle de Hopfield correspondent à un apprentissage non-supervisé : le réseau se corrigera lui-même jusqu'à atteindre un état stable.

Le modèle de Hopfield a pour but de résoudre des problèmes d'optimisation et de retrouver des patterns à partir de clés proposées en input en tenant comptes d'éventuelles dégradations ou du bruit qu'a pu subir le modèle proposé en input : on parle de mémoire associative. Son apprentissage se fait selon la loi de Hebb.

Donald Hebb (1904-1985), neuropsychologue canadien, conçoit dans les années 1950 une théorie à propos de l'apprentissage neuronal : pour lui, l'activation répétée d'une synapse en modifie la force, le « poids », et donc permet de faire circuler l'information plus vite d'un neurone à l'autre. Cette théorie s'applique aussi pour les neurones artificiels, puisqu'elle a été retranscrite presque telle qu'elle pour eux, en étant simplement numérisée. La loi de Hebb, numériquement, est la suivante :

$$w_{ij} = \frac{1}{p} \sum_{k=1}^p x_i^k x_j^k$$

où w_{ij} est le poids de la connexion entre les neurones i et j , p le nombre de patterns appris, et x_i^k la k ième input pour le neurone i (de même pour le neurone j). Dans le modèle de Hopfield, si $i = j$, la connexion w_{ij} est égale à 0. La connexion w_{ij} est symétrique ($w_{ij} = w_{ji}$).

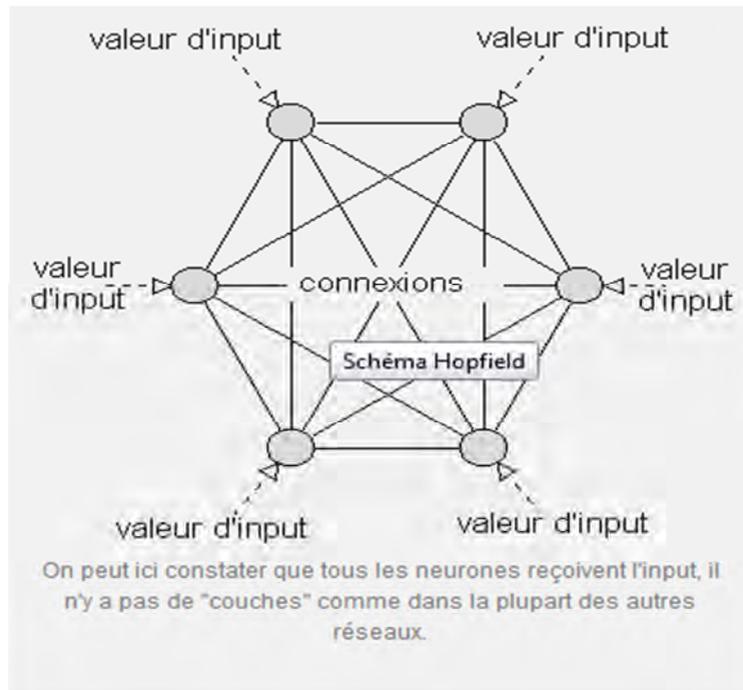


Figure (2-4) : modèle de Hopfield [17]

Le modèle de Hopfield est la version la plus simple des « réseaux neuronaux ». Elle est proposée en 1982 par le physicien John Hopfield. Il s'agit d'une structure dite complètement connectée puisque chaque neurone est connecté à tous les autres. Ainsi, le neurone ne possède ni couche d'entrée, ni couche de sortie. Les neurones ont un état binaire (classiquement, 1 ou -1, même si certains modèles utilisent 1 et 0).

La première étape de l'apprentissage est la mémorisation des pattern par le réseau. Contrairement à la loi de Hebb classique, où la modification des poids synaptiques se fait par période, le modèle de Hopfield ne modifie ces poids qu'une fois que tous les pattern lui ont été présentés. On lui présente ensuite un vecteur en input. Le réseau va alors modifier le poids des connexions de certains neurones (aléatoirement), et itérera cette action jusqu'à convergence :

1. On présente au neurone un vecteur d'input.
2. Le réseau calcule l'output pour l'input donnée.
3. Si la sortie n'est pas la sortie souhaitée, le réseau modifie le poids de certaines connexions selon la formule suivante : $\Delta w_{ij} = \Delta w_{ji} = R \cdot (a_i \cdot a_j)$ où R est une constante positive non-nulle représentant la force d'apprentissage ; a_i et a_j étant

respectivement les valeurs d'activation des neurones i et j ; et Δw_{ij} et Δw_{ji} étant égales et représentant la valeur de la connexion synaptiques entre les neurones i et j .

Le réseau retourne ensuite à l'étape 1 jusqu'à ce que la sortie souhaitée soit obtenue.

Néanmoins, le modèle de Hopfield possède une limite très claire : pour n unités dans le réseau, il ne peut mémoriser que $0,14n$ patterns. Au delà, on observe un phénomène appelé « oubli catastrophique » : la reconnaissance du réseau devient alors hautement chaotique. Un scientifique du nom de Hoffman a ainsi établi un parallèle entre le comportement du réseau après un oubli chaotique et celui d'un patient schizophrène.

La méthode d'apprentissage non-supervisé du réseau de Hopfield permet de le rapprocher d'un enfant découvrant le monde autour de lui. Ce réseau modifie ses poids synaptiques en conséquences de ce qu'il constate, de la même façon que nos propres neurones. Il s'agit donc d'une modélisation très simplifiée de notre mémoire notamment. Néanmoins, Hopfield n'avait pas pour but de mimer le comportement d'un neurone biologique, mais plutôt de tenter de minimiser l'énergie des réseaux en prouvant que les réseaux interconnectés pouvaient consommer une énergie décroissante.

Par ailleurs, la loi de Hebb est une loi s'appliquant en premier lieu aux neurones humains : ce fonctionnement est d'abord un fonctionnement biologique avant d'être celui de neurones formels. En effet, lors de l'apprentissage humain, si une synapse est très sollicitée de part et d'autre, les deux neurones et elles vont « apprendre » et permettre que l'information passe plus vite : au bout d'un moment, pour obtenir la même intensité d'activation du second neurone, le premier aura besoin de moins d'intensité lui-même.

2.1.4. La levée des limitations

Les machines de Boltzmann (1983) : La Machine de Boltzmann est le premier modèle connu apte à traiter de manière satisfaisante les limitations recensées dans le cas du perceptron. Mais l'utilisation pratique s'avère difficile, la convergence de l'algorithme étant extrêmement longue (les temps de calcul sont considérables) [15] .

Les machines de Boltzmann sont un réseau de neurones récurrent et multicouches.

Ce fut l'un des premier réseau de neurone capable d'apprendre des représentation interne et de résoudre des problème combinatoire difficile.

Les connexions des machines de Boltzmann sont généralement restreintes pour pouvoir résoudre efficacement des problèmes pratiques.

Forme des machines de Boltzmann

Forme générale

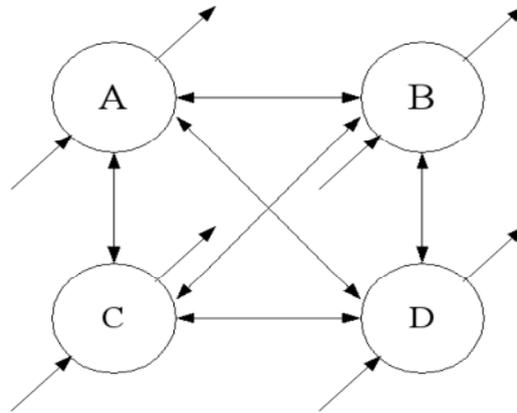


Figure (2-5) : Forme des machines de Boltzmann

Les unités (ou neurones) sont reliés par des liaisons pondérées par des poids. chaque neurone peut prendre deux états: actif ou inactif.

Il y a deux types de neurones: les neurones visibles et les neurones cachés.

Les neurones visibles sont utilisés comme entrée ou comme sortie.

Les neurones cachés servent à rendre le réseau plus complexe et donc plus efficace.

Les connexions d'une machine de Boltzmann ont deux restrictions:

- $w_{ii} = 0$ i . (pas de connexion avec soi même)
- $w_{ij} = w_{ji}$ i, j . (toutes les connexions sont symétriques) où w_{ij} est le poids de connexion entre le neurone i et j .

Tous les neurones sont connectés entre eux.

Machine de Boltzmann restreinte

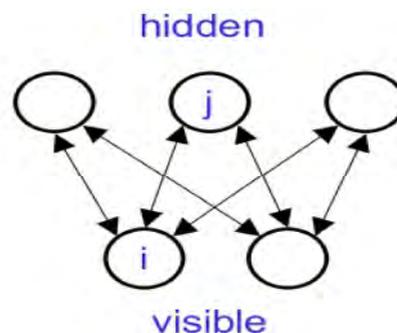


Figure (2-6): Machine de Boltzmann restreinte

Les machines de Boltzmann restreintes ajoutent des contraintes au réseau pour rendre l'apprentissage plus efficace.

Il n'y a pas de connexions à l'intérieur d'une même couche de neurones.

Énergie

Les machines de Boltzmann sont un réseau avec une "énergie" définie pour le réseau:

$$E = - \sum_{i < j} w_{ij} s_i s_j + \sum_i \theta_i s_i$$

où:

- w_{ij} est le poids de connexion entre le neurone i et j.
- s_i est l'état du neurone i (1 ou 0)
- θ_i est le biais du neurone i.

La différence dans l'énergie globale qui résulte de l'activation d'un neurone i noté E_i , est donné par :

$$E_i = \sum_j w_{ij} s_j - \theta_i$$

Équilibre Thermodynamique

Une machine de Boltzmann est faite d'unités stochastiques. La probabilité, p_i que le i ème neurone s'active est donné par:

$$p_i = \frac{1}{1 + \exp\left(-\frac{1}{T} E_i\right)}$$

Où le scalaire T est défini comme la température du système.

Le réseau fonctionne en choisissant un neurone à chaque fois et en modifiant son état selon la formule ci-dessus.

Si les neurones sont activés séquentiellement dans un ordre qui ne dépend pas de leurs entrées, le réseau finira par atteindre une distribution de Boltzmann (aussi appelé sa distribution d'équilibre ou stationnaire) dans laquelle la probabilité d'un vecteur d'état v est déterminé

uniquement par l'énergie du vecteur relativement à l'énergie de tous les autres vecteurs d'état possibles.

$$P(v) = \frac{e^{-E(v)}}{\sum_u e^{-E(u)}}$$

2.2. Apprentissage

Étant donné un ensemble de vecteurs d'états d'entraînement (les données), l'apprentissage consiste à trouver les poids et les biais (les paramètres) qui définissent une distribution de Boltzmann dans laquelle les vecteurs d'entraînements ont une forte probabilité. Notre fonction objectif est donc l'espérance de la log-probabilité des vecteurs v , où v est dans l'ensemble d'entraînement.

2.2.1. Rétropropagation du gradient

En informatique, la technique de rétropropagation du gradient (Backpropagation en anglais) est une méthode qui permet de calculer le gradient de l'erreur pour chaque neurone d'un réseau de neurones, de la dernière couche vers la première. De façon abusive, on appelle souvent technique de rétropropagation du gradient, l'algorithme classique de correction des erreurs basé sur le calcul du gradient grâce à la rétropropagation et c'est cette méthode qui est présentée ici. En vérité, la correction des erreurs peut se faire selon d'autres méthodes, en particulier le calcul de la dérivée seconde. Cette technique consiste à corriger les erreurs selon l'importance des éléments qui ont justement participé à la réalisation de ces erreurs. Dans le cas des réseaux de neurones, les poids synaptiques qui contribuent à engendrer une erreur importante se verront modifiés de manière plus significative que les poids qui ont engendré une erreur marginale.

Ce principe fonde les méthodes de type algorithme du gradient, qui sont efficacement utilisées dans des réseaux de neurones multicouches comme les perceptrons multicouches (MLP pour « multi-layers perceptrons »). L'algorithme du gradient a pour but de converger de manière itérative vers une configuration optimisée des poids synaptiques. Cet état peut être un minimum local de la fonction à optimiser et idéalement, un minimum global de cette fonction (dite fonction de coût).

Normalement, la fonction de coût est non linéaire au regard des poids synaptiques. Elle dispose également d'une borne inférieure et moyennant quelques précautions lors de

l'apprentissage, les procédures d'optimisation finissent par aboutir à une configuration stable au sein du réseau de neurones.

2.2.1.1. Historique

Les méthodes de rétropropagation du gradient firent l'objet de communications dès 1975 (Werbos), puis 1985 (Parker et LeCun), mais ce sont les travaux de Rumelhart, Hinton & Williams en 1986 qui suscitèrent le véritable début de l'engouement pour cette méthode [16].

2.2.1.2. Utilisation au sein d'un apprentissage supervisé

Dans le cas d'un apprentissage supervisé, des données sont présentées à l'entrée du réseau de neurones et celui-ci produit des sorties. La valeur des sorties dépend des paramètres liés à la structure du réseau de neurones : connectique entre neurones, fonctions d'agrégation et d'activation ainsi que les poids synaptiques.

Les différences entre ces sorties et les sorties désirées forment des erreurs qui sont corrigées via la rétropropagation, les poids du réseau de neurones sont alors changés. La manière de quantifier cette erreur peut varier selon le type d'apprentissage à effectuer. En appliquant cette étape plusieurs fois, l'erreur tend à diminuer et le réseau offre une meilleure prédiction. Il se peut toutefois qu'il ne parvienne pas à échapper à un minimum local, c'est pourquoi on ajoute en général un terme d'inertie (momentum) à la formule de la rétropropagation pour aider l'algorithme du gradient à sortir de ces minimums locaux.

➤ Algorithme

Les poids dans le réseau de neurones sont au préalable initialisés avec des valeurs aléatoires (val a). On considère ensuite un ensemble de données qui vont servir à l'apprentissage. Chaque échantillon possède ses valeurs cibles qui sont celles que le réseau de neurones doit à terme prédire lorsqu'on lui présente le même échantillon. L'algorithme se présente comme ceci :

Soit un échantillon \vec{x} que l'on met à l'entrée du réseau de neurones et la sortie recherchée pour cet échantillon \vec{t}

- On propage le signal en avant dans les couches du réseau de neurones :

$$x_k^{(n-1)} \quad x_j^{(n)}$$

- La propagation vers l'avant se calcule à l'aide de la fonction d'activation g , de la fonction d'agrégation h (souvent un produit scalaire entre les poids et les entrées du neurone) et des poids synaptiques $\overrightarrow{w_{jk}}$ entre le neurone $x_k^{(n-1)}$ et le neurone $x_j^{(n)}$. Attention au passage à cette notation qui est inversée, $\overrightarrow{w_{jk}}$ indique bien un poids de k vers j .

$$x_j^{(n)} = g^{(n)} \left(h_j^{(n)} \right) = g^{(n)} \left(\sum_k w_{jk}^{(n)} x_k^{(n-1)} \right)$$

- Lorsque la propagation vers l'avant est terminée, on obtient à la sortie le résultat \vec{y}

On calcule alors l'erreur entre la sortie donnée par le réseau \vec{y} et le vecteur \vec{t} désiré à la sortie pour cet échantillon. Pour chaque neurone i dans la couche de sortie, on calcule :

$$e_i^{sortie} = g'(h_i^{sortie})(t_i - y_i)$$

(g' est la dérivée de g)

- On propage l'erreur vers l'arrière e_i^n e_j^{n-1} grâce à la formule suivante :

$$e_j^{(n-1)} = g'^{(n-1)}(h_j^{(n-1)}) \sum_i w_{ij} e_i^{(n)}$$

note:
$$e_j^{(n)} = \sum_i [t_i - y_i] \frac{\partial y_i}{\partial h_j^n}$$

- On met à jour les poids dans toutes les couches :

$w_{ij}^{(n)} = \lambda e_i^{(n)} x_j^{(n-1)}$ où λ représente le taux d'apprentissage (de faible magnitude et inférieur à 1.0)

Implémentation :

L'algorithme présenté ici est de type « online », c'est-à-dire que l'on met à jour les poids pour chaque échantillon d'apprentissage présenté dans le réseau de neurones. Une autre méthode est dite en « batch », c'est-à-dire que l'on calcule d'abord les erreurs pour tous les échantillons sans mettre à jour les poids (on additionne les erreurs) et lorsque l'ensemble des données est

passé une fois dans le réseau, on applique la rétropropagation en utilisant l'erreur totale. Cette façon de faire est préférée pour des raisons de rapidité et de convergence.

L'algorithme est itératif et la correction s'applique autant de fois que nécessaire pour obtenir une bonne prédiction. Il faut cependant veiller aux problèmes de sur apprentissage liés à un mauvais dimensionnement du réseau ou un apprentissage trop poussé.

Ajout d'inertie :

Pour éviter les problèmes liés à une stabilisation dans un minimum local, on ajoute un terme d'inertie (momentum). Celui-ci permet de sortir des minimums locaux dans la mesure du possible et de poursuivre la descente de la fonction d'erreur. À chaque itération, le changement de poids conserve les informations des changements précédents. Cet effet de mémoire permet d'éviter les oscillations et accélère l'optimisation du réseau. Par rapport à la formule de modification des poids présentée auparavant, le changement des poids avec inertie au temps t se traduit par :

$$w_{ij}^{(n)}(t) = \lambda e_i^{(n)} x_j^{(n-1)} + \alpha w_{ij}^{(n)}(t-1)$$

avec α un paramètre compris entre 0 et 1.0.

De nos jours, les réseaux multicouches et la rétropropagation de gradient reste le modèle le plus étudié et le plus productif au niveau des applications.

2.3. Les Réseaux De Neurones Appliqués Aux Génie Civil

Les réseaux de neurones artificiels simulent très étroitement le système nerveux biologique. Ce sont des modèles de calcul parallèle pour la représentation et le traitement de l'information. Du fait de leur grande similitude avec le cerveau, les réseaux de neurones ont des capacités semblables à celle de l'être humain et notamment ; l'apprentissage, la classification et la simulation ou prédiction.

Les réseaux de neurones sont des modèles de calcul très robustes et tolérants aux informations bruitées ou même incomplètes, ces caractéristiques ont fait des réseaux de neurones un bon candidat pour une large variété d'applications en génie civil. Parmi les premiers articles publiés dans ce domaine c'est celui de Adeli et Yeh paru en 1989 décrivant l'utilisation et l'apprentissage du perceptron dans la conception en ingénierie, et depuis plusieurs équipes se sont orientés vers l'application des réseaux de neurones aux problèmes de mécanique des

structures, modélisation des matériaux, contrôle des vibrations, prédiction de fissures, détection des dommages, régulation des transports et la liste est longue

2.3.1. Application des RNA en analyse des structures

Les réseaux de neurones à rétropropagation sont beaucoup utilisés en analyse et conception des structures, comportement des matériaux et identification des dommages. Ils sont utilisés pour représenter la relation force-déplacement en analyse statique des structures (Hajela et Berke 1991), la conception des structures en béton armé (Sanad et Saka 1996), la détection des dommages directement après séisme (Wong et al. 1996) et la prévision du coût et de la durée de vie des structures dans une phase de préconception (Sdhabhon 1998).

2.3.2. Application des RNA en géotechnique

Plusieurs systèmes ont été développés pour la caractérisation des sites, la classification des sols et des roches, le calcul des fondations, la liquéfaction des sols, et le calcul des barrages (Teh et al. 1997, Ural et Saka 1999).

2.3.3. Application des RNA en technologie des bétons

Dans le domaine des matériaux bon nombres d'applications ont vu le jour, surtout dans les domaines de la modélisation, de la composition des bétons, de la durabilité des matériaux composites de la fissuration etc. le point de départ dans la modélisation des matériaux est toujours un ensemble de résultats expérimentaux. Dans la technique de modélisation analytique traditionnelle, le comportement du matériau est observé dans le but d'identifier ses caractéristiques, une fois chose faite, un modèle mathématique est développé pour simuler ce comportement. Ce processus consiste à coder la connaissance du comportement dans un ensemble de règles mathématiques.

La technique de modélisation par les réseaux de neurones introduite par J. Ghaboussi et son équipe Wu et Garrett en 1991, utilise la capacité d'apprentissage de ces derniers pour simuler le comportement du matériau.

A partir des tests de laboratoire, le réseau de neurones subit la première phase d'apprentissage, qui fait que l'information est codée et stockée dans le poids des interconnexions. La deuxième phase est celle de la simulation (validation), dans cette phase les données de quelques tests non-utilisés dans la phase d'apprentissage sont pris et comparés aux résultats obtenus par le réseau de neurones. Les auteurs ont trouvé que cette technique

donnait de très bons résultats. Tous ça sans avoir recours à développer des règles ou utiliser des approximations analytiques.

Souvent les tests sont conçus pour que l'état de contrainte en un point soit constant. Sidarta et Ghaboussi ont publié des travaux sur la modélisation des matériaux en géotechniques en introduisant de nouveaux tests faisant inclure un état de contrainte non-uniforme. En utilisant les réseaux de neurones ils ont montré qu'on pouvait, à partir d'un seul test, extraire plus d'informations qu'à partir de plusieurs tests traditionnels. Peu de temps après les mêmes auteurs ont utilisé un nouveau type de réseaux de neurones pour l'appliquer à la modélisation du béton, sable et matériaux composites. Ce nouveau type est ce qu'on appelle

Nested Adaptive Neural Network (NANN), qui prend avantage de la structure inclusive de donnée du milieu étudié. Le principe est d'utiliser un ancien réseau de neurones, lui ajouter de nouveaux modules et refaire une partie de l'apprentissage, comme ça on obtient un nouveau réseau de neurones représentant le comportement du milieu étudié dans une dimension plus grande. L'application de cette technique a été le chemin de contrainte dans l'essai triaxial.

Plus récemment Hashash et al ont proposé l'utilisation des réseaux de neurones (NANN) dans la modélisation par les éléments finis des problèmes de frontière. Sachant que les réseaux de neurones ont l'avantage d'apprendre continuellement suivant la disponibilité des données, et qu'ils ne nécessitent pas d'intégrations spéciales pour leur implémentation dans l'analyse par éléments finis contrairement aux modèles élastoplastiques.

2.3.4. Composition des bétons

Le béton est l'un des matériaux de construction les plus utilisés dans le monde, il est composé d'éléments bien connus ciment, eau, fines, agrégats etc. Dans la composition et la qualité du béton, la résistance à la compression axiale est une des propriétés les plus importantes.

D'autres propriétés telles que le module d'élasticité, la perméabilité, la résistance aux agents atmosphériques etc. sont directement liées à cette résistance. Seulement cette résistance ne peut être obtenue qu'après le 28ème jour de fabrication, des erreurs expérimentales sont inévitables et à cette date il est souvent trop tard d'apporter des améliorations si les tests sont insatisfaisants. C'est pourquoi il est fortement désirable de connaître une estimation de la résistance avant la fabrication du béton. Plusieurs études ont été menées pour déterminer la résistance du béton en utilisant les réseaux de neurones

Kasperkiewicz et al en 1995 ont utilisé les réseaux de neurones artificiels pour déterminer la composition d'un béton à haute performance en utilisant la technique de

Carpenter et al des réseaux de neurones a résonance adaptative.

Oh et al en 1999 ont employé un RNA comme outil pour minimiser les incertitudes et les erreurs de conception des proportions des mélanges de béton). D'autres ont établi un système d'acquisition de connaissance basé sur les RNA pour la conception d'un mélange de béton (Ji-Zong al. 1999). Nehdi et al. ont étudié les performances des bétons auto-compactés en prévoyant leurs propriétés par un système de RNA à base de l'algorithme Levenberg-Marquadt de Rétropropagation (Nehdi et al. 2001). Cette étude présente la première tentative pour prévoir les propriétés du béton-auto-nivelant (BAN) frais et durci rendues complexes par les phénomènes de ségrégation et de ressuage. Dias et Pooliyada ont proposé un modèle RNA à rétropropagation avec des proportions constitutives du béton comme entrées et des propriétés du béton comme sorties (Dias et Pooliyada 2001).

Ce modèle a été employé pour prédire la résistance à la compression et l'ouvrabilité d'un béton prêt à l'emploi (BPE) contenant des retardateurs et d'un béton à haute résistance (BHP) contenant des adjuvants chimiques et/ou des ajouts minéraux.

Plusieurs chercheurs ont publié peut de temps après des articles sur l'utilisation des réseaux de neurones pour la prédiction de la résistance d'une composition de béton.

La technique de rétropropagation back propagation neural network (BPNN) a été utilisée les premiers temps, laissant place à d'autres techniques plus récente [25] comme les réseaux de neurones probabilistes (PNN) améliorant ainsi la qualité et le temps d'apprentissage.

2.3.5. Hydratation du ciment

Il existe plusieurs modèles de RNA pour prévoir les propriétés de durcissement de la pâte de ciment comme le degré d'hydratation, le taux d'évolution de la chaleur d'hydratation, l'humidité relative et la porosité totale (Basma et al. 1999, Parka et al. 2005).

2.3.6. Retrait et fluage des bétons

Très peu de travaux portent sur le retrait et le fluage des bétons. Karthikeyan et al. ont utilisés les RNA pour la prédiction du fluage et du retrait des bétons à hautes Performances

(Karthikeyan et al. 2008). Maru et Nagpal ont aussi utilisés les RNA pour la simulation des déformations de retrait et de fluage du béton armé (Maru et Nagpal 2004).

2.3.7. Durabilité et autres :

L'approche RNA est utilisée pour prévoir la profondeur de carbonatation du béton (Yun-Hong et Bin 2004), l'identification des coefficients de perméabilité (Shouju et al. 2002), la diffusion et la pénétration des chlorures (Peng et al. 2002), et la dégradation par les acides sulfuriques à diverses concentrations (Hewayde et al. 2007).

Waszczyszyn et al ont utilisé la technique de régularisation des réseaux de neurones associée à l'analyse des problèmes mal posés pour l'estimation de la fatigue du béton. Gotlib et al ont utilisé les réseaux de neurones pour l'analyse des matériaux composites hétérogènes désordonnés. L'utilisation des réseaux de neurones pour l'évaluation des paramètres de la distribution statistique a été faite par Fairbairn et al dans le but de simuler la fissuration du béton par la méthode des éléments finis. L'étude de la fissuration d'une poutre en béton armé renforcée par des fibres dans les zones de fissuration peut être faite par la MEF, néanmoins, cela implique beaucoup de complication, une technique empirique par les réseaux de neurones a été utilisé par Flood dans le but de faciliter cette étude. D'autres utilisations dans la conception des éléments en acier ont également été faites dans les références. Et la liste reste longue et ouverte.

2.4. Avantages des RNA

Les RNA fonctionnent sans programme, n'exécutent pas d'instructions, et ne manipulent pas de nombres. La destruction d'une partie de ces circuits n'empêche pas le réseau de fonctionner. Ces principales propriétés sont :

- Parallélisme : c'est l'exécution de plusieurs tâches (données numériques et non pas symboliques) en même temps, ce qui explique la grande rapidité de calcul des RNA.
- Capacité d'apprentissage : elle permet aux réseaux de tenir compte des contraintes et des données provenant de l'extérieur. Elle se caractérise dans certains réseaux par leur capacité d'auto-organisation qui assure leur stabilité.
- Généralisation : elle permet au réseau de trouver une solution généralisée applicable à tous les exemples du problème, même s'ils contiennent des erreurs ou sont inachevés ou ne sont pas présentés pendant l'apprentissage.
- Mémoire distribuée : la mémoire est répartie sur plusieurs entités et sur tout le réseau.

3. MODELISATION PAR RESEAUX DE NEURONES ARTIFICIELS (RNA)

3.1. Introduction

L'approche réseaux de neurones artificiels (RNA) est bien adaptée pour traiter les systèmes complexes. Cette approche est issue des travaux réalisés dans le domaine de l'intelligence artificielle et sont inspirés du fonctionnement du cerveau humain en se basant principalement sur le concept de neurone biologique [19, 20, 21, 22, 23]. Chaque auteur a défini les réseaux de neurones à sa manière. La plus classique [24] considère les RNA comme un processus massivement distribué en parallèle, qui présente une propension naturelle pour stocker de la connaissance empirique et la rendre disponible à l'usage. Il ressemble au cerveau sur deux aspects :

- La connaissance est acquise par le réseau à travers un processus d'apprentissage.
- Les connexions entre les neurones (poids synaptiques) servent à stocker la connaissance.

Il existe une grande variété d'agencements possibles de neurones artificiels [25], mais le type le plus utilisé pour la prévision est le perceptron multicouche à rétropropagation de l'erreur ([16] [26], [27] [28]). Ce réseau comporte une couche d'entrées, une ou plusieurs couches cachées (multi layers perceptron) et une couche de sorties. Chaque couche contient des unités de calcul (neurones) connectées à d'autres neurones par des poids.

3.2. Réseaux Neurons Multicouche à Rétropropagation

3.2.1. Définition

Ce modèle est une extension du modèle perceptron, il est sans doute le plus simple et le plus connu des réseaux de neurones. C'est un réseau multicouche (Multi Layer Perceptron MLP) appelé également Perceptron multicouches. Il est capable de résoudre toute opération logique complexe. Ce réseau utilise le modèle d'apprentissage supervisé où l'on fournit au réseau un ensemble d'exemples, chaque exemple est constitué d'un vecteur d'entrée (valeurs numériques) et de son vecteur de sortie désirée. Les poids sont au départ aléatoires, et c'est par un mécanisme « d'essai erreur correction » que ce type de réseau évolue vers un état stable. Cela porte le nom de règle Rétropropagation du Gradient (minimisation d'une fonction dépendante de l'erreur).

Cette méthode est la plus utilisée pour l'entraînement des réseaux multicouches statiques. Elle a été développée par le groupe de recherche Parallel Distributed Processing (PDP).

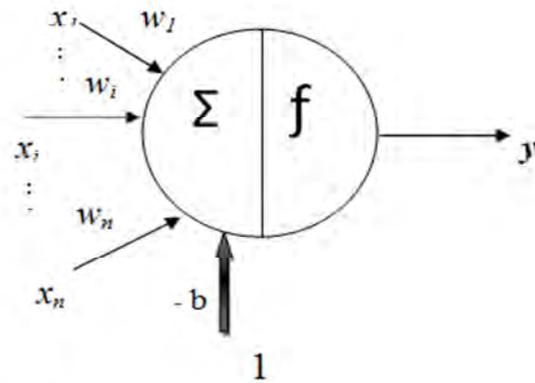


Figure (3-1) Structure d'un neurone artificiel

$x = \{x_1, x_2, \dots, x_i, \dots, x_n\}$ vecteur d'entrées

$w = \{w_1, w_2, \dots, w_i, \dots, w_n\}$ Vecteur des poids synaptiques

b seuil

$$\text{net} = \sum_{i=1}^n w_i x_i$$

$$v = \text{net} - b$$

$y = f(v)$ fonction d'activation ou fonction de transfert

3.2.2. Structure et fonctionnement

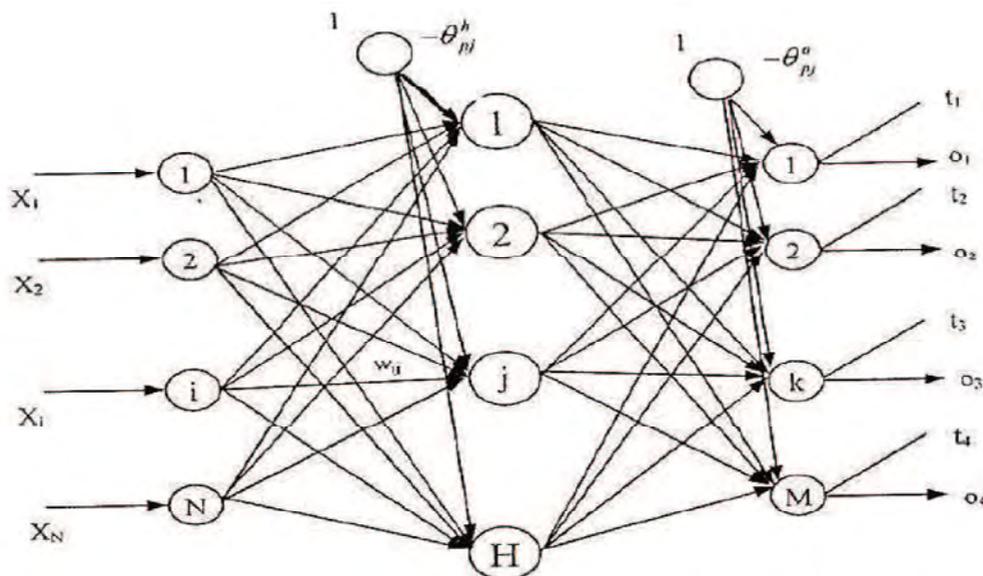


Figure (3-2) Topologie d'un réseau de neurones multicouche à rétropropagation.

Sur la figure :

$$x = \{x_1, x_2, \dots, x_i, \dots, x_N\}$$

Vecteur des entrées du réseau de neurones de dimension N ;

$$o = \{o_1, o_2, \dots, o_i, \dots, o_M\}$$

Vecteur des sorties du réseau de neurones de dimension M ;

$$t = \{t_1, t_2, \dots, t_i, \dots, t_M\}$$

Vecteur cible du réseau de neurones de dimension M ;

W_{ji}^h poids synaptique connectant le nœud j dans la $h^{\text{ième}}$ couche cachée au nœud i de la couche cachée précédente ou la couche d'entrée.

W_{kj}^o Poids synaptique connectant le nœud k dans la couche de sortie au nœud j de la dernière couche cachée.

θ_{pj}^h Valeur du seuil du $p^{\text{ième}}$ vecteur de la base d'apprentissage, $j^{\text{ième}}$ nœud de la $h^{\text{ième}}$ couche cachée.

θ_{pk}^o Valeur du seuil du $p^{\text{ième}}$ vecteur de la base d'apprentissage, $k^{\text{ième}}$ nœud de la couche de sortie.

Le fonctionnement des réseaux de neurones à rétropropagation peut être divisé en deux étapes : introduction des entrées et rétropropagation de l'erreur. Durant ce processus, les poids synaptiques sont constants. A la fin de la première itération, le vecteur de sortie est comparé au vecteur voulu, et une valeur d'erreur est calculée pour chaque nœud de sortie. Ce signal d'erreur est propagé dans le sens inverse de la couche de sortie aux nœuds de la couche cachée adjacente, chaque neurone reçoit une part de cette erreur selon sa contribution au vecteur de sortie, c'est-à-dire selon son poids synaptique. Ce processus est répété à chaque itération, couche par couche, où les poids synaptiques vont être mis à jour, jusqu'à ce que le réseau converge vers un état stable.

3.2.3. Phase d'apprentissage

L'apprentissage est supervisé, c'est-à-dire qu'on associe une configuration d'entrée à une configuration de sortie. L'algorithme de rétropropagation est utilisé pour l'apprentissage du réseau. Cet algorithme est un gradient itératif conçu pour minimiser l'erreur quadratique moyenne (MSE) entre la sortie obtenue et la sortie désirée. Cette minimisation est réalisée par une configuration des poids adéquate. L'erreur (e) est la différence entre la valeur désirée (t) pour le neurone de sortie et sa valeur calculée par propagation (o).

3.2.3.1. Introduction des données

Lorsque le $p^{\text{ième}}$ exemple de la base d'apprentissage est fourni à la couche d'entrée, la valeur

du nœud j de la couche cachée est :

$$\text{net}_{pj}^h = \sum_{i=1}^n w_{ji}^h \cdot x_i - \theta_{pj}^h$$

La sortie de ce nœud est :

$$o_{pj}^h = f_j^h (\text{net}_{pj}^h)$$

De façon similaire, la sortie du nœud k de la couche de sortie est :

$$\text{net}_{pk}^o = \sum_{j=1}^M w_{kj}^o \cdot o_{pj}^h - \theta_{pk}^o$$

$$o_{pk} = f_k^o (\text{net}_{pk}^o) = f_k^o \left(\sum_{j=1}^M w_{kj}^o \cdot o_{pj}^h - \theta_{pk}^o \right)$$

ou f_j^h et f_k^o représentent les fonctions de transfert du $j^{\text{ième}}$ nœud de la couche cachée

et du nœud k de la couche de sortie. Ces fonctions doivent être continues et différentiable.

3.2.3.2. Calcul de l'erreur

L'erreur commise entre le vecteur de sortie du réseau de neurone et le vecteur désiré est calculée comme suit :

$$\delta_{pk} = t_{pk} - o_{pk}$$

$$E_p = \frac{1}{2} \sum_{k=1}^M \delta_{pk}^2 = \frac{1}{2} \sum_{k=1}^M (t_{pk} - o_{pk})^2$$

L'algorithme de la rétropropagation applique une correction $\Delta W(p)$ aux poids synaptiques, qui est proportionnelle au gradient $\partial E_p / \partial W(p)$ selon l'équation suivante :

$$W(p+1) = w(p) + \Delta W(p) = w(p) - \mu \frac{\partial E_p}{\partial w(p)}$$

ou $0 < \mu < 1$ est un paramètre d'apprentissage. Puisque le gradient $\partial E_p / \partial W(p)$ est différent entre les nœuds de la couche de sortie et la couche cachée, la formule de correction des poids synaptiques sera différente.

3.2.3.3. Modification des poids synaptiques liants la couche de sortie

Le gradient de E_p par rapport à w_{kj}^o est calculé comme suit :

$$\mu \frac{\partial E_p}{\partial w_{kj}^o} = (t_{pk} - o_{pk}) \frac{\partial f_k^o}{\partial \text{net}_{pk}^o} \frac{\text{net}_{pk}^o}{w_{kj}^o} = (t_{pk} - o_{pk}) o_{pj}^h \frac{\partial f_k^o}{\partial \text{net}_{pk}^o}$$

On pose le gradient local du nœud k :

$$\delta_{pk}^o = (t_{pk} - o_{pk}) \frac{\partial f_k^o}{\partial \text{net}_{pk}^o}$$

Les valeurs des poids synaptiques des nœuds connectés avec la couche de sortie sont :

$$w_{kj}^o(p+1) = w_{kj}^o(p) + \Delta w_{kj}^o = w_{kj}^o(p) + \mu \delta_{pk}^o o_{pj}^h$$

3.2.3.4. Modification des poids synaptiques liants la couche cachée

L'erreur totale E_p est reliée aux poids synaptiques, liés à la couche cachée par la formule :

$$E_p = \frac{1}{2} \sum_{k=1}^M \delta_{pk}^2 = \frac{1}{2} \sum_{k=1}^M (t_{pk} - o_{pk})^2 = \frac{1}{2} \sum_{k=1}^M (t_{pk} - f_k^o \left[\sum_{j=1}^M w_{kj}^o \cdot o_{pj}^h - \theta_{pk}^o \right])^2$$

Le gradient de E_p par rapport aux poids synaptiques liés à la couche cachée est

$$\begin{aligned} \frac{E_p}{w_{ji}^h} &= \frac{1}{2} \sum_{k=1}^M \frac{(t_{pk} - o_{pk})^2}{w_{ji}^h} \\ &= - \sum_{k=1}^M (t_{pk} - o_{pk}) \frac{f_k^o}{net_{pk}^o} \frac{net_{pk}^o}{o_{pj}^h} \frac{o_{pj}^h}{net_{pj}^h} \frac{net_{pj}^h}{w_{ji}^h} \\ &= - \sum_{k=1}^M (t_{pk} - o_{pk}) w_{kj}^o \cdot x_i \frac{f_k^o}{net_{pk}^o} \frac{f_j^h}{net_{pj}^h} \\ &= - \frac{f_j^h}{net_{pj}^h} x_i \sum_{k=1}^M \delta_{pk}^o w_{kj}^o \end{aligned}$$

On pose le gradient local du nœud j de la couche cachée :

$$\delta_{pj}^h = \frac{f_j^h}{net_{pj}^h} \sum_k \delta_{pk}^o w_{kj}^o$$

Donc

$$w_{ji}^h(p+1) = w_{ji}^h(p) + \delta_{pj}^h x_i = w_{ji}^h(p) + \mu \delta_{pj}^h x_i$$

Il est à noter que les termes d'erreurs pour les unités cachées doivent être calculés avant la mise à jour des poids synaptiques des unités de sorties.

Cette phase d'apprentissage ou d'entraînement exige la compréhension complète du problème. Il est essentiel ici d'identifier les paramètres d'entrée et de sortie. Une fois que la décision concernant la structure du réseau est prise, l'apprentissage comporte l'acquisition de connaissances, la détermination du nombre de couches cachées et le nombre de neurones dans chaque couche. Le nombre optimal de ces derniers est déterminé par essai afin de réaliser la meilleure exécution pour le réseau.

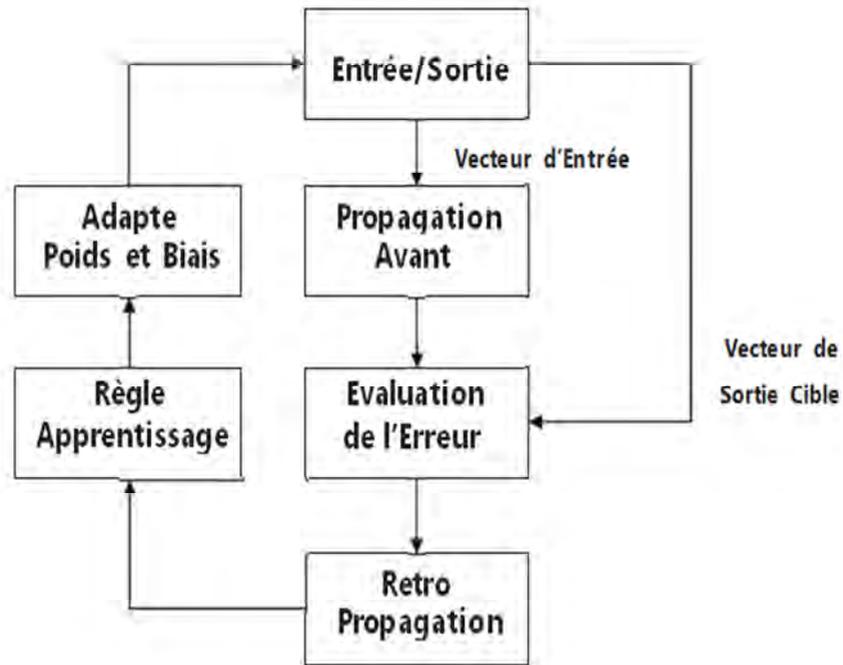


Figure (3-3) Organigramme de Rétropropagation [29] .

3.2.4. Phase de reconnaissance (test) et de validation

Avant qu'un réseau de neurones puisse être employé avec n'importe quel degré de confiance.

Il faut établir la validité des résultats qu'il produit. C'est la phase d'utilisation du réseau ou phase de test. Il existe une autre phase dite de validation, souvent conduite avec la phase de test. Habituellement les phases de test et de validation impliquent d'évaluer la performance du réseau sur des données qui n'ont pas été employées pour la phase d'apprentissage. Les solutions désirées et celles produites par le réseau peuvent être comparées d'une façon qualitative, ou d'une façon quantitative en utilisant un test statistique.

3.2.5. Performances du réseau de neurones multicouche à rétropropagation

La performance d'un modèle de réseau de neurones peut être évaluée en utilisant différentes expressions, notamment :

- Le coefficient de détermination R^2

$$R^2 = \frac{\sum_{i=1}^n (t_i - \bar{t})^2}{\sum_{j=1}^M (y_j - \bar{y})^2}$$

- La Somme de l'erreur quadratique (SSE : Sum of squared error)

$$SSE = \sum_{i=1}^n (t_i - o_i)^2$$

- L'erreur quadratique moyenne (MSE : Mean Square Error)

$$MSE = \frac{1}{n} \sum_{i=1}^n (t_i - o_i)^2$$

avec

t_i : la sortie désiré du modèle i ; o_i : la sortie réelle du modèle i ; et N : le nombre total de données.

Le coefficient de détermination R^2 représente la proportion de variation de la valeur de sortie qui est « expliquée » par le modèle, sa valeur idéale est 1.

Le nombre de nœuds dans la couche cachée n'est pas déterminé à priori. On détermine la performance du réseau à l'aide du facteur (MSE) pour un nombre croissant de neurones de la couche cachée. Le nombre retenu correspond à celui qui minimise l'erreur quadratique.

4. MODELES RESEAUX DE NEURONES SUR LE MORTIER

4.1. Introduction

Le réseau de neurones multicouche à rétropropagation a été choisi dans cette étude en raison de sa popularité et de sa capacité de réussite en termes de classification, de prédiction et d'optimisation des modèles. L'emploi de l'apprentissage supervisé exige la connaissance d'une sortie désirée pour tout élément de l'ensemble des données d'entrée. Le modèle multicouche à rétropropagation se compose de trois types de couches, la couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Dans la prévision du résistance à la compression du mortier, l'ensemble des données d'entrée est représenté par les compositions chimique et minéralogique du ciment, Cela représente les paramètres de la couche d'entrée qui sont transmis à la couche cachée. Enfin, la couche de sortie reçoit son entrée de la couche cachée. Dans notre cas la sortie de cette couche représente les résistances à la compression du mortier.

4.2. Méthodologie et mise en œuvre

La première étape consiste à identifier les paramètres à employer pour les modèles de prévision. Les données sont divisées en trois parties. 70% des données sont employées pour l'apprentissage, 15% pour la phase de test et 15% pour la validation. La mise en œuvre d'un modèle RNA inclut ces trois phases : l'apprentissage, le test et la validation .

La phase d'apprentissage ou d'entraînement a pour but de déterminer les paramètres de connexion du réseau à l'aide de la technique d'optimisation. La phase de test consiste à vérifier le réseau déterminé lors de la phase d'apprentissage sur des données non utilisées lors de cette dernière, et à examiner la capacité du réseau à généraliser les exemples d'apprentissage (en comparant la sortie réelle du réseau avec la sortie désirée). La phase de validation est effectuée sur la dernière partie des données

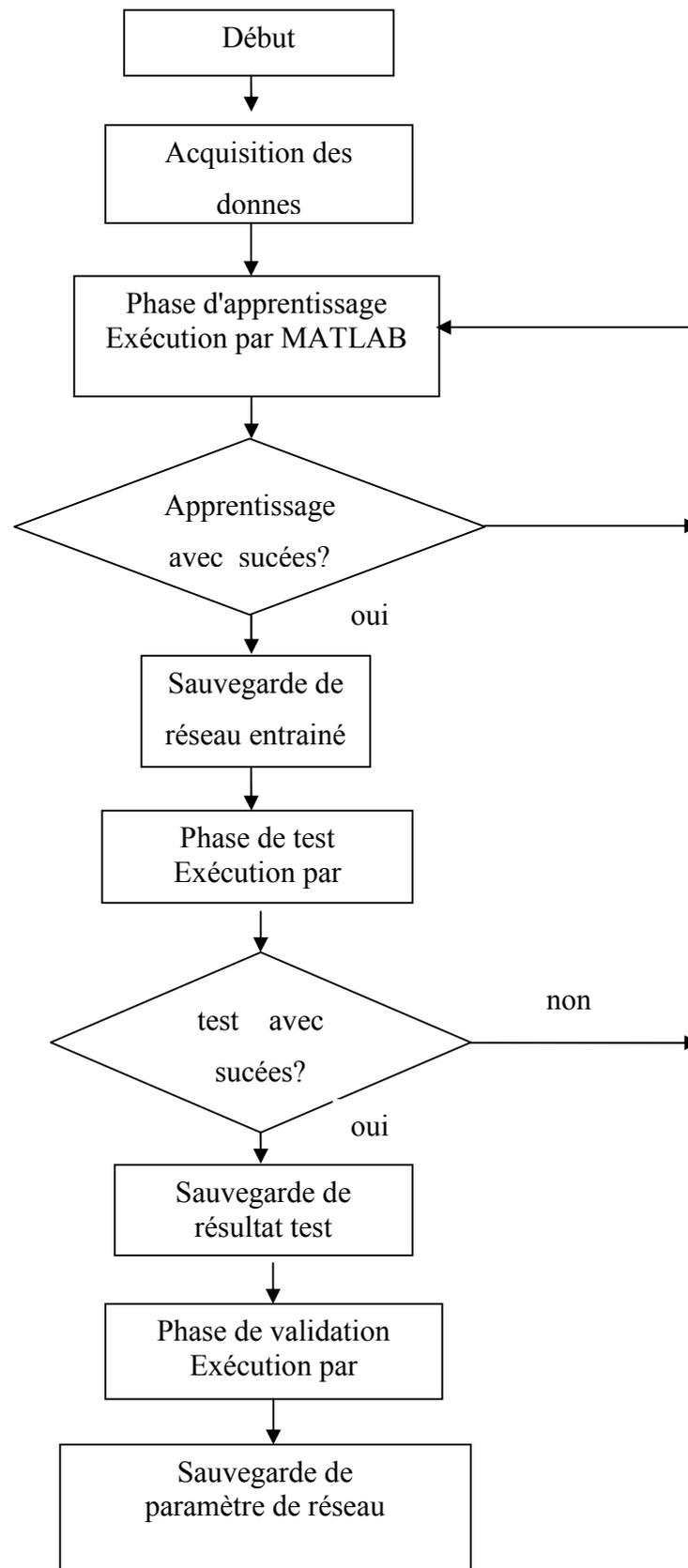


Figure (4-1) Méthodologie de développement du Réseau de neurones artificiels (RNA).

4.3.Langage de programmation utilisé

Différents programmes de simulation par réseaux de neurones sont disponibles et commercialisés (Neurone Solution, Brain Maker, Matlab ToolBoxes, Statistica Neural Network, etc.). Le choix du simulateur se fait sur la base des avantages et des inconvénients de chacun d'eux. Le coût, la flexibilité d'utilisation, le type de plate forme et l'ergonomie de l'interface sont les principaux critères pour évaluer chaque simulateur. Pour des raisons économiques, nous avons choisi le logiciel MATLAB 7.12.0 (R2011a) .

Ce logiciel (MATLAB 2011) a été développé pour être utilisé dans un environnement efficace et flexible de simulation pour la recherche et l'application des RNA, soit généralement par le développement d'un programme ou l'utilisation de l'application NETWORK DATA MANAGER. Cette dernière est développée pour fonctionner sous Windows avec une interface utilisateur graphique facile à utiliser. Elle est aussi employée pour créer, manœuvrer, et visualiser les réseaux de neurones d'une part et importer, exporter les données et les résultats via ces réseaux d'autres part. Pour illustration, le panneau principal de NETWORK DATA MANAGER est présenté Fig. 4.2.

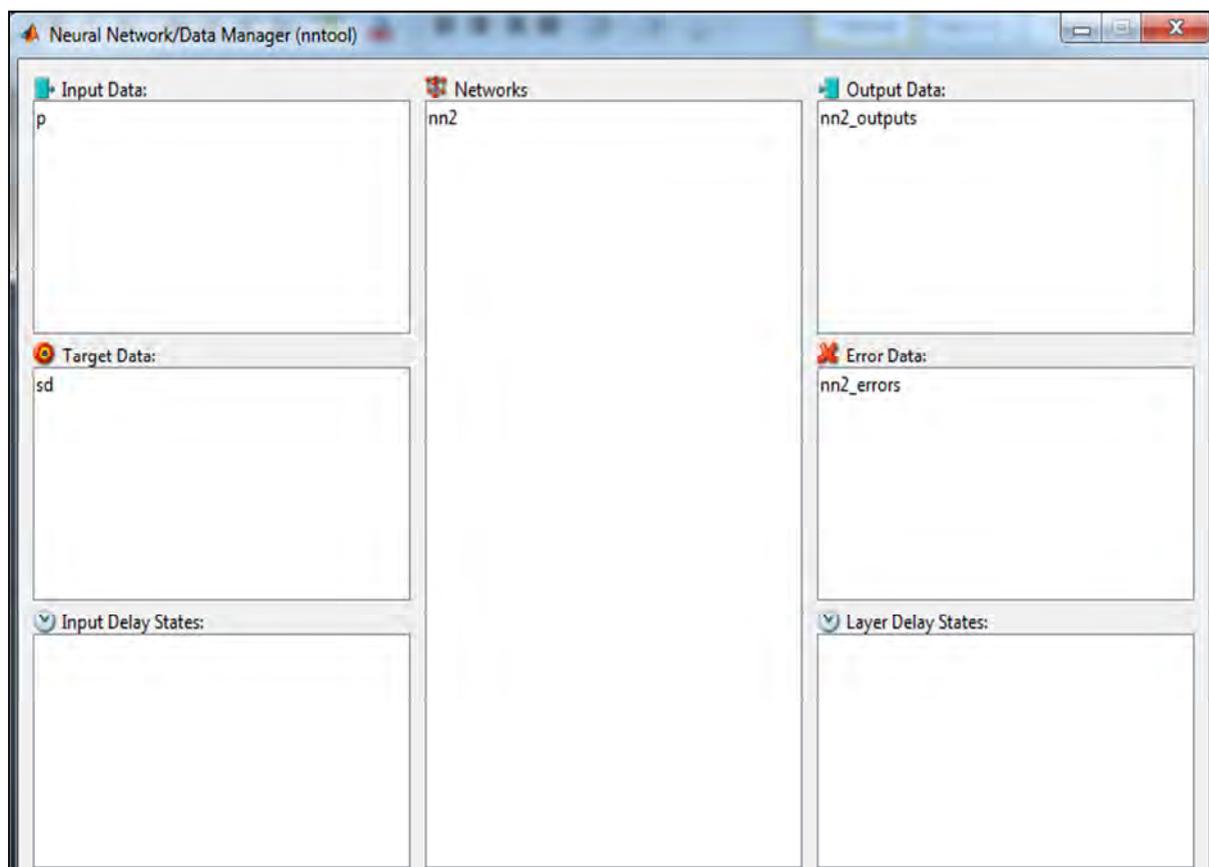


Figure (4-2) Panneau principal de l'application NETWORK DATA MANAGER.

La première étape consiste à déclarer la matrice des entrées et des sorties sous la forme d'un fichier Excel. La normalisation des données (d'entrée et de sortie) après traitement est nécessaire afin qu'ils prennent des valeurs entre -1 et 1 pour être en accord avec la fonction de transfert Tan-Sigmoïde utilisée.

La seconde étape est la phase d'apprentissage, qui consiste à initialiser les coefficients de corrélation des trois phases (apprentissage, test et validation), puis à initialiser le nombre de neurones par couche cachée, sachant que nous avons testé une couche cachée, puis deux couches cachées, en créant une boucle qui s'incrémente de 1 à 50 neurones pour la première couche cachée, et de même pour la seconde couche.

La création d'un nouveau réseau de neurones (net) qui contient des entrées et des sorties normalisées, un certain nombre de couche cachées et de neurones par couche, passe par l'utilisation d'une fonction de transfert correspondant à chaque couche (Tansig : couche cachée ; Purelin : couche de sortie), d'un algorithme d'apprentissage Trainlm qui utilise les techniques de rétropropagation Levenberg-Marquardt (Levenberg-Marquardt backpropagation).

Pour entrainer le réseau il faut déclarer divers paramètres, comme le nombre de cycle pendant l'apprentissage (itérations ; epochs), la fréquence de vérification de l'erreur d'apprentissage vis-à-vis de l'erreur désirée (goal), le taux d'apprentissage (η : mu). Il faut aussi déclarer la division des données en trois parties selon les trois phases (70% pour l'apprentissage, 15% pour le test et 15% pour la validation). Le cycle d'apprentissage signifie un passage complet par tous les neurones.

Après la phase d'apprentissage, on passe à la simulation du réseau en utilisant la décomposition des données entre test et validation. Après l'exécution de ce programme on obtient deux fichiers. Le premier comprend les informations sur l'historique de l'apprentissage et donne les mesures statistiques de la performance du réseau, comme les coefficients de corrélation et l'erreur quadratique moyenne pour les trois phases et pour chaque réseau de neurones créé. Le second inclut les valeurs des poids de connexion (synaptiques) et les biais entre les neurones pour chaque réseau de neurones créé (il faut souligner qu'on commence avec un réseau qui contient une seule couche cachée et un seul neurone, et on passe progressivement à un réseau qui contient deux couches cachées de 50 neurones chacune).

4.4.Elaboration des modèles RNA

4.4.1. Introduction

Ce sous-chapitre comporte la présentation des trois groupes de modèles (chacun contient deux modèles), le 1^{er} contient les deux modèles de résistance à la compression du mortier en fonction de la composition chimique et l'autre de la fonction minéralogique du clinker et la 2^{ème} contient les deux modèles de l'expansion de la pâte du clinker suivant leur composition chimique et minéralogique et le dernier contient les deux modèles de la consistances de la pâte du clinker suivant leur composition chimique et minéralogique .

Un ensemble de données y compris 631 échantillons de données provenant d'étude expérimentale ont été utilisés pour les réseaux de neurones artificiels. De ceux-ci, 425 modèles de données ont été utilisés pour la formation du réseau, et le reste (206 modèles), ont été choisis au hasard et utilisés comme l'ensemble de données de test des données d'apprentissage qui sont indiqués dans l'ANNEXE .

4.4.2.Modèle Global du Résistance a la compression

i. Analyse Chimique

Le tableau récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations.

Tableau (4-1): Indicateurs utilisés – plage de variation [30].

Type de paramètres	Description	Plage de variation
Entrées	SiO_2 : Silice	$20.3 < SiO_2 < 22.6$
	Al_2O_3 : Alumine	$5.06 < Al_2O_3 < 5.79$
	Fe_2O_3 : Oxyde ferrique	$2.7 < Fe_2O_3 < 3.52$
	c_{a0} : la chaux	$62.8 < c_{a0} < 66.6$
	M_{g0} : Oxyde de magnésium	$1.48 < M_{g0} < 1.87$
	k_{20} : Oxyde de potassium	$0.64 < k_{20} < 0.89$
	Na_{20} : Oxyde de sodium	$0.09 < Na_{20} < 0.27$
	J : les jours	2 j , 7 j , 28 j
Sorties	Résistance à la compression Du mortier	$12 MPa < R_c < 53 MPa$

Dans un premier temps, on procède à la détermination du type d'algorithme d'apprentissage à utiliser dans notre étude. Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère le plus efficace et le plus fiable.

Le tableau montre une comparaison entre différents algorithmes d'apprentissage en termes de nombre d'itérations maximale, de temps nécessaire pour la convergence et de degré de performance atteint. Dans ce cas on a fixé 1000 itérations comme étant la limite du processus de convergence.

Tableau (4-2): Comparaison entre les algorithmes d'apprentissage.

Algorithme	Technique	Nombre d'itération	Temps (sec)	performance
Train oss	One step secant	150	02	0.992
Train br	Bayesian regulation	56	00:58	0.966
Train gd	Descente gradient	107	01	1.52
Train gdm	Descente gradient avec moneutum	82	01	2
Train gda	Gradient descente with adaptive learning rate	124	01:08	0.982
Train bfg	BFGS Quasi newton	115	02:02	0.487
Train lm	Levenberg Marquardt	64	01:56	0.321

Une fois l'algorithme d'apprentissage choisi, on procède à la détermination du nombre de couches cachées et de neurones par couche cachée. Ceci a été effectué en faisant des simulations numériques avec différentes architectures du réseau (en faisant varier le nombre de neurones de chaque couche cachée).

Tableau (4-3): Résumé de l'effet du nombre de couches cachées.

Réseaux de neurones	Fonction de transfert	Nombre d'itération	Performance
RN 8-10-5-1	Tansig-Tansig-Purelin	24	1.98
RN 8-25-15-1	Tansig-Tansig-Purelin	100	1.40
RN 8-40-20-1	Tansig-Tansig-Purelin	88	0.688
RN 8-50-30-1	Tansig-Tansig-Purelin	64	0.321

Les résultats semblent montrer une légère différence dans l'exécution avec le changement de la taille de la couche cachée pour les deux premières architectures. Les réseaux avec plus de neurones ont légèrement surpassé ceux avec un nombre moyen de neurones dans les couches

cachées. Autrement dit, les erreurs associées diminuent dans des grands réseaux beaucoup plus rapidement que dans des réseaux à nombre moyen de neurones. Basé sur les observations ci-dessus, l'analyse des résultats obtenus à partir des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de taille RN 8-50-30-1 pour le modèle de prévision du résistance à la compression du mortier.

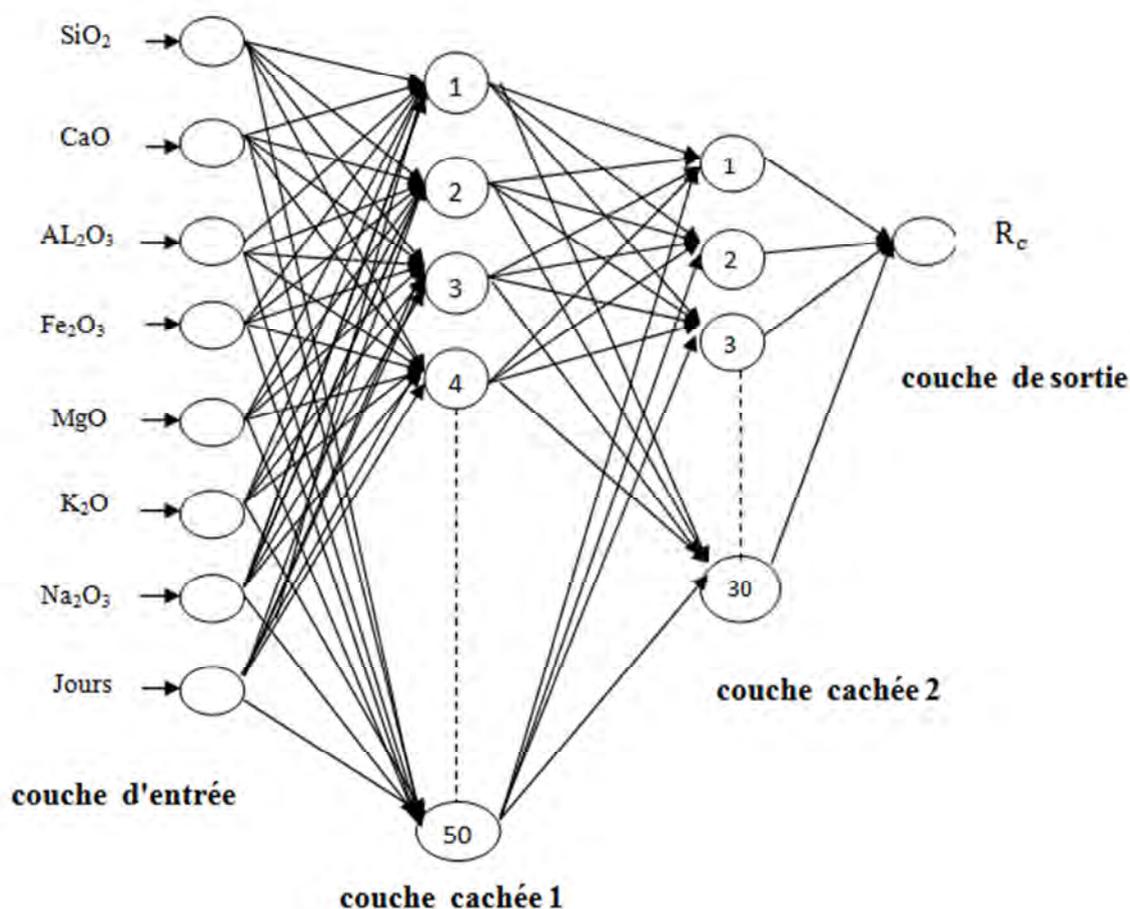
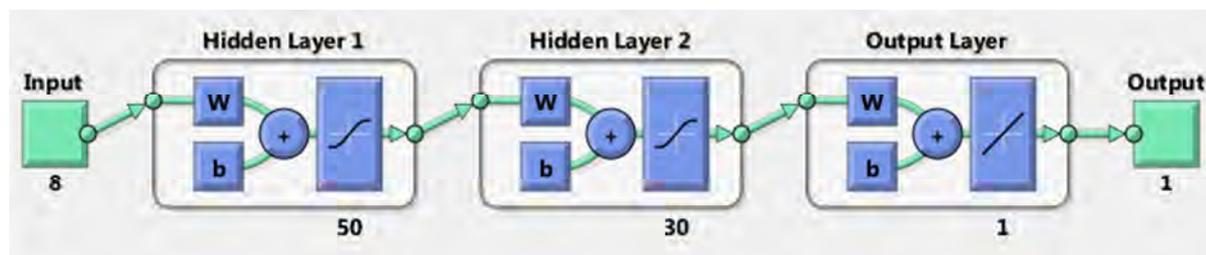


Figure (4-3) Architecture adoptée pour la Résistance en fonction de la composition chimique de clinker.

Au début du processus d'apprentissage, tous les poids de connexion ont été aléatoirement initialisés entre -1 et +1. Le nombre maximum de cycles ou itérations a été fixé à 1000 pour le modèle de réseau de neurone global de prévision du résistance (MRNGPR)..

Tous les exemples d'entrée ont été présentés dans un ordre aléatoire dans chaque cycle. L'apprentissage a été basé sur l'algorithme de Levenberg-Marquardt à Rétropropagation avec les fonctions de transfert standard Tansigmoïde et Purelinéaire correspondant respectivement aux couches cachées et de sortie, avec un taux d'apprentissage de 0.01 .

Le tableau suivant récapitule les différents paramètres d'apprentissage et l'architecture du réseau de neurone à rétropropagation adapté à notre application.

Tableau (4-4): Architecture et paramètres de MRNGPR

Paramètres	Architecture du MRNGPR				Paramètres du MRNPR			Ensembles des données	
	N.PE	N.PS	N.CC	N.N.CC	N.ITR	η	E2	Entrées	Sortie
Modèle réseaux de neurone global de la prévision du résistance (MRNGPR) (8-50-30-1)	8	1	2	50-30	1000	0.01	0.00	SiO_2 , Al_2O_3 , Fe_2O_3 , c_aO , MgO , k_2O , Na_2O , J	Rc

(N.PE: Nombre de paramètres d'entrée, N.PS : Nombre de paramètres de sortie, N.CC : Nombre de couches cachées, N.N.CC : Nombre de neurones dans chaque couche cachée, N.ITR : Nombre d'itérations).

le tableau 4-5 et la figure 4-4 ,la figure 4-5 donnent les résultats des déférents phases d'application de modèle MRNGPR .

Tableau 4-5 : performance de modèle MRNGPR

Modèle RNA	Apprentissage	
	R2	MSE
MRNGPR(8-50-30-1)	0.9677	0.321

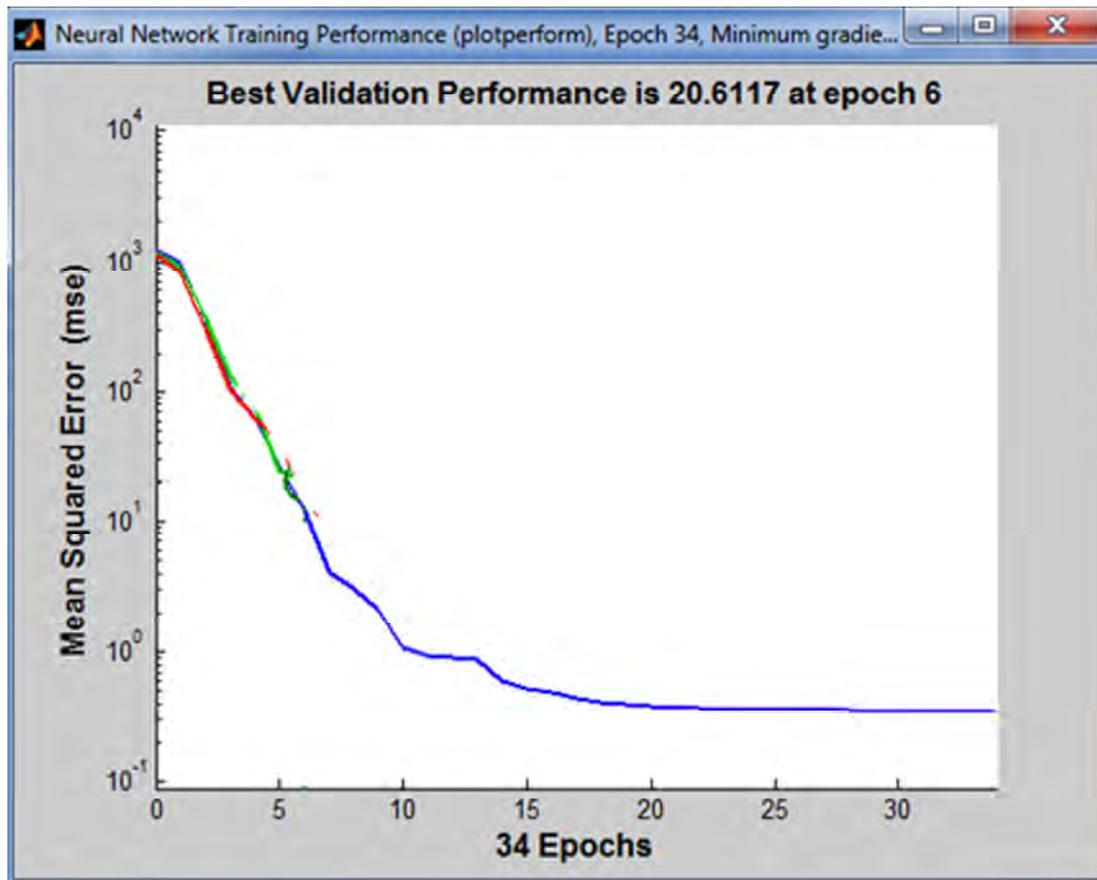
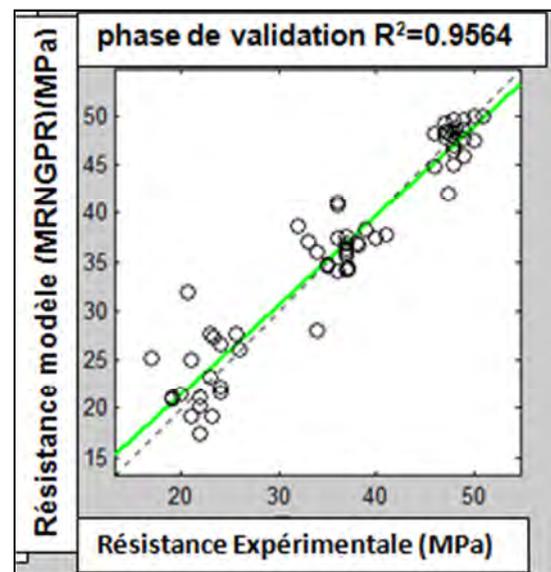
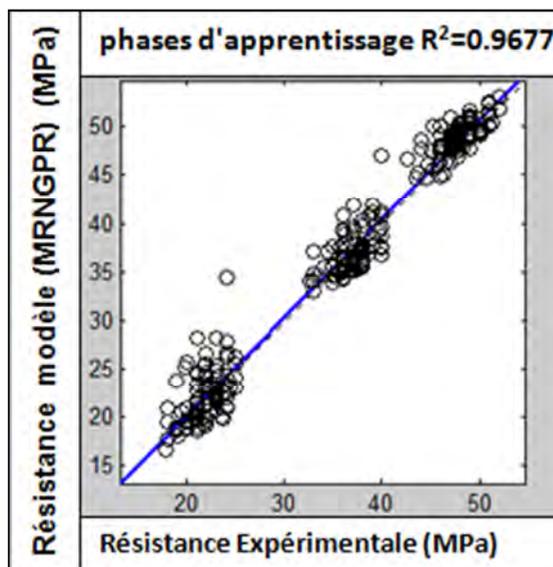


Figure (4-4) Apprentissage du modèle MRNGPR



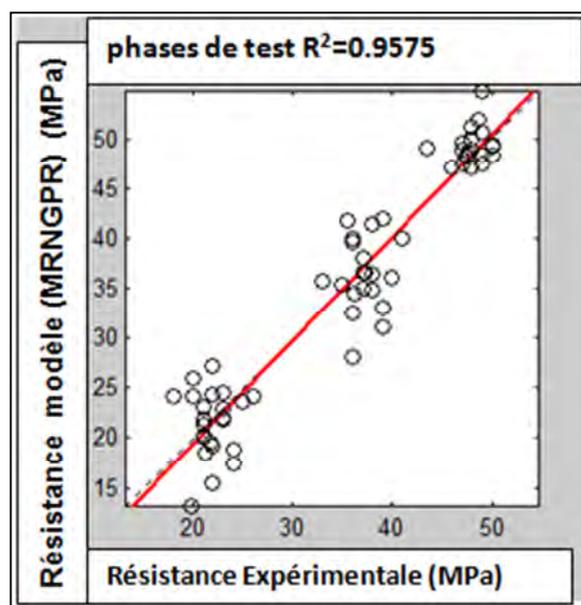


Figure (4-5) Performance du Modèle Global MRNGPR(la composition chimique).

ii. Analyse minéralogique

la composition chimique du cru ainsi que le régime de cuisson et de refroidissement du clinker sont les facteurs principaux qui déterminent la composition minéralogique du ciment portland, pour pouvoir estimer la composition minéralogique on a introduit des formules de BOGUE et on a trouvé les données qui sont exposé a l'ANNEXE .

Le tableau (4-6) récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations.

Tableau (4-6): Indicateurs utilisés – plage de variation.

Type de paramètres	Description	Plage de variation
Entrées	C ₃ S:silicate tricalcique	49 < C ₃ S < 65.49
	C ₂ S:silicate bicalcique	10.95 < C ₂ S < 28.38
	C ₃ A: aluminate tricalcique	7.89 < C ₃ A < 10.21
	C ₄ AF : alumino ferrite tetracalcique	8.23 < C ₄ AF < 11.82
	J:jours	2j , 7j , 28j
Sorties	Résistance à la compression Du mortier	12 MPa < R _c < 53 MPa

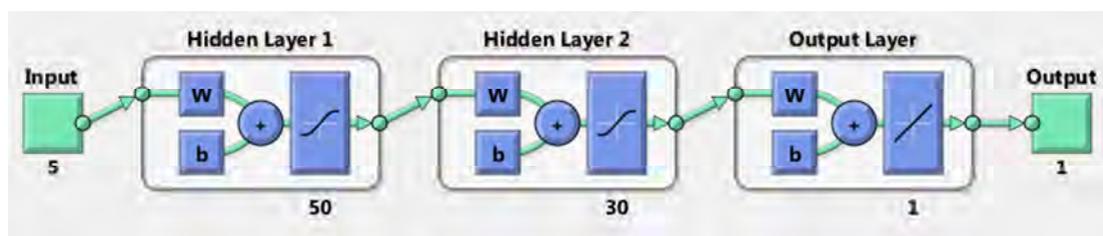
Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère ici aussi le plus efficace et le plus performant.

on procède à la détermination du nombre de couches cachées et de neurones par couche cachée. Ceci a été effectué en faisant des simulations numériques avec différentes architectures du réseau (en faisant varier le nombre de neurones de chaque couche cachée).

Tableau (4-7): Résumé de l'effet du nombre de couches cachées.

Réseaux de neurones	Fonction de transfert	Nombre d'itération	Performance
RN 5-7-4-1	Tansig-Tansig-Purelin	158	0.682
RN 5-25-10-1	Tansig-Tansig-Purelin	100	0.566
RN 5-40-15-1	Tansig-Tansig-Purelin	79	0.458
RN 5-50-30-1	Tansig-Tansig-Purelin	124	0.245

L'analyse des résultats obtenus lors des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de taille RN 5-50-30-1 pour le modèle de prévision du résistance.



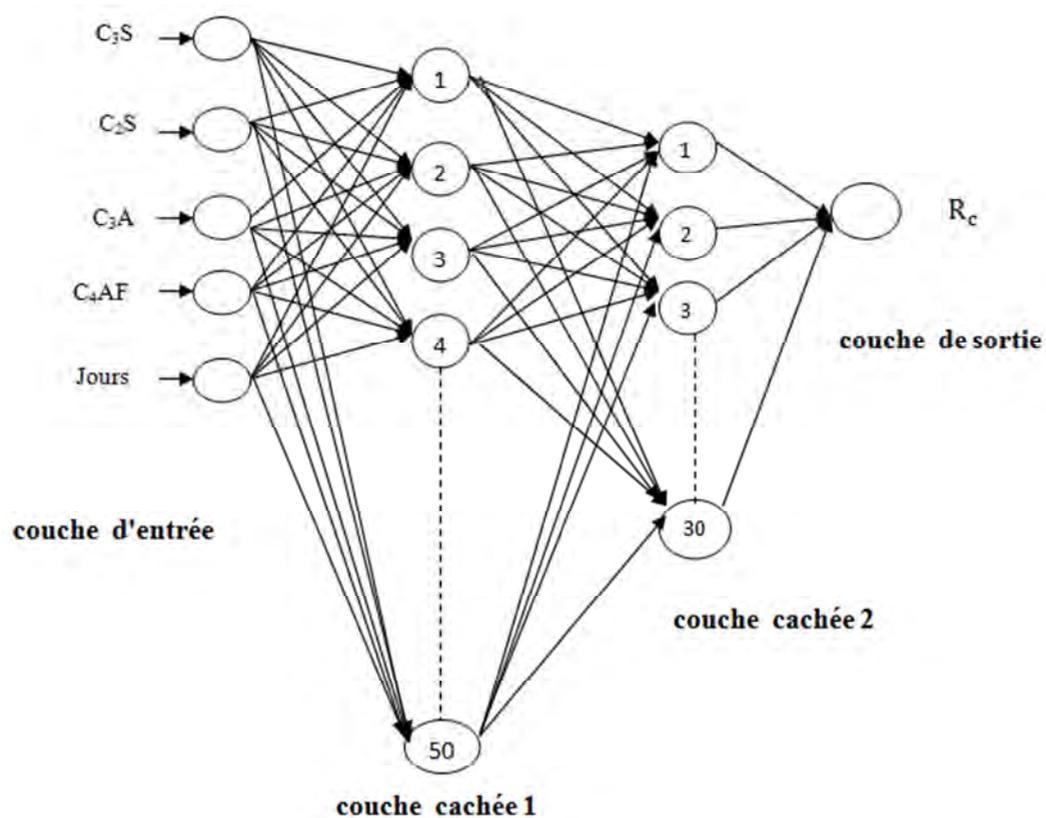


Figure (4-6): Architecture adoptée pour la Résistance.

Le tableau 4-8 et la figure 4-7 , la figure 4-8 donnent les résultats des différents phases d'application de modèle MRNGPR (la composition minéralogique).

Tableau (4-8) : performance de modèle MRNGPR (la composition minéralogique).

Modèle RNA	Apprentissage	
	R2	MSE
MRNGPR(5-50-30-1)	0.9903	0.245

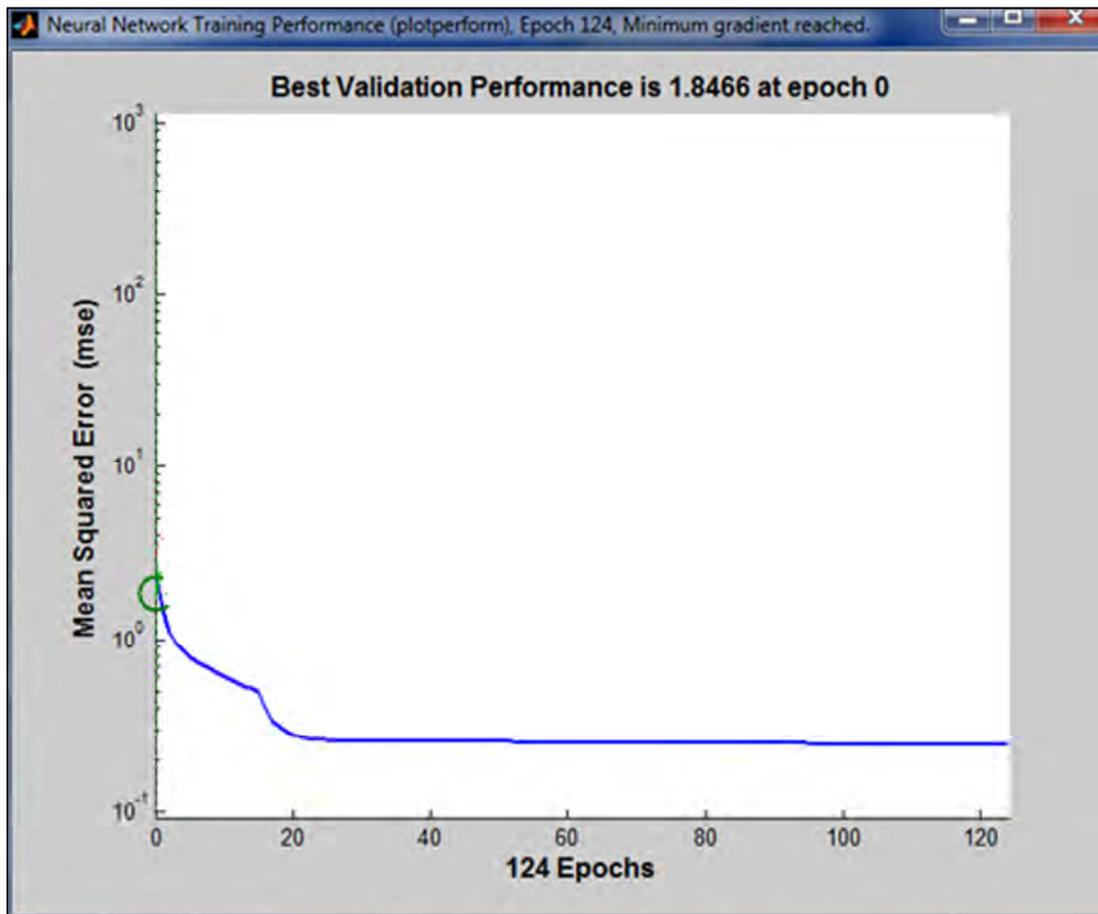
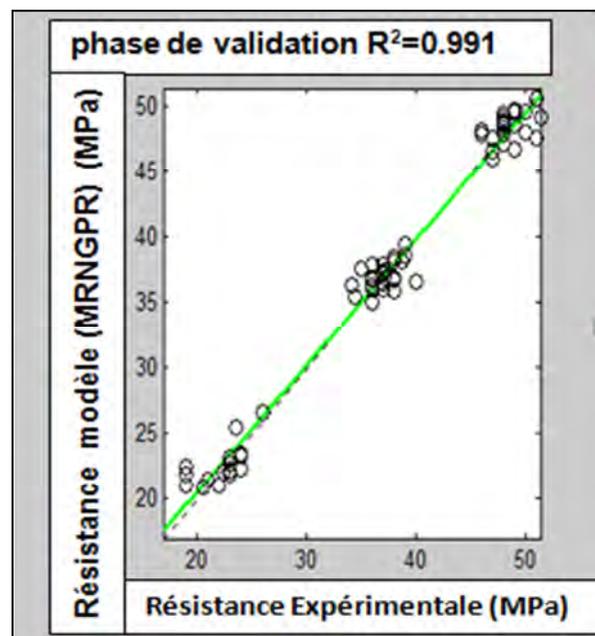
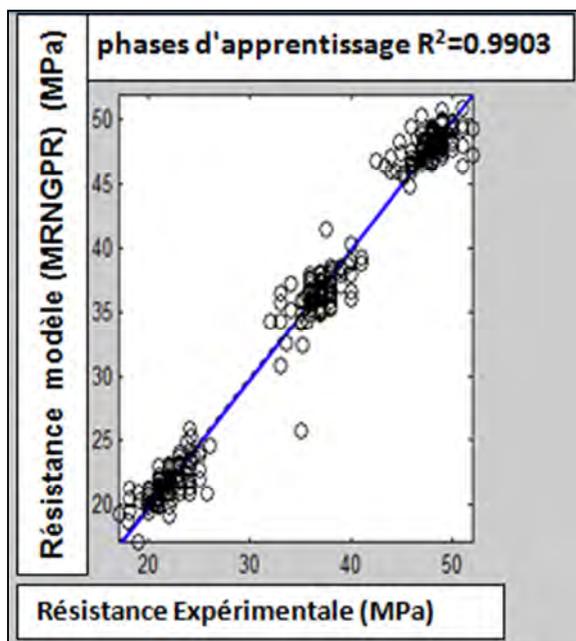


Figure (4-7) Apprentissage de modèle MRNGPR (la composition minéralogique)



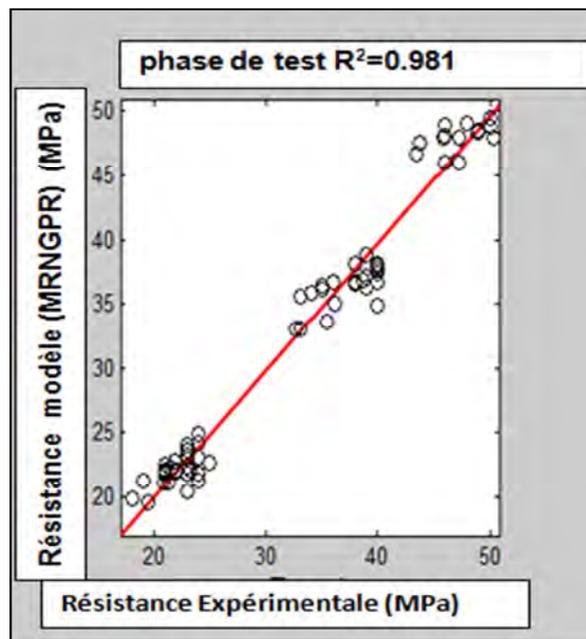


Figure (4-8) Performance du Modèle Global MRNGPR (la composition minéralogique).

iii. Etude paramétrique

Le but de cette étude paramétrique est de quantifier l'effet de certains de ces composantes en utilisant le modèle de réseau de neurones (MRNGPR).

a) Influence des alcalis (K_2O) :

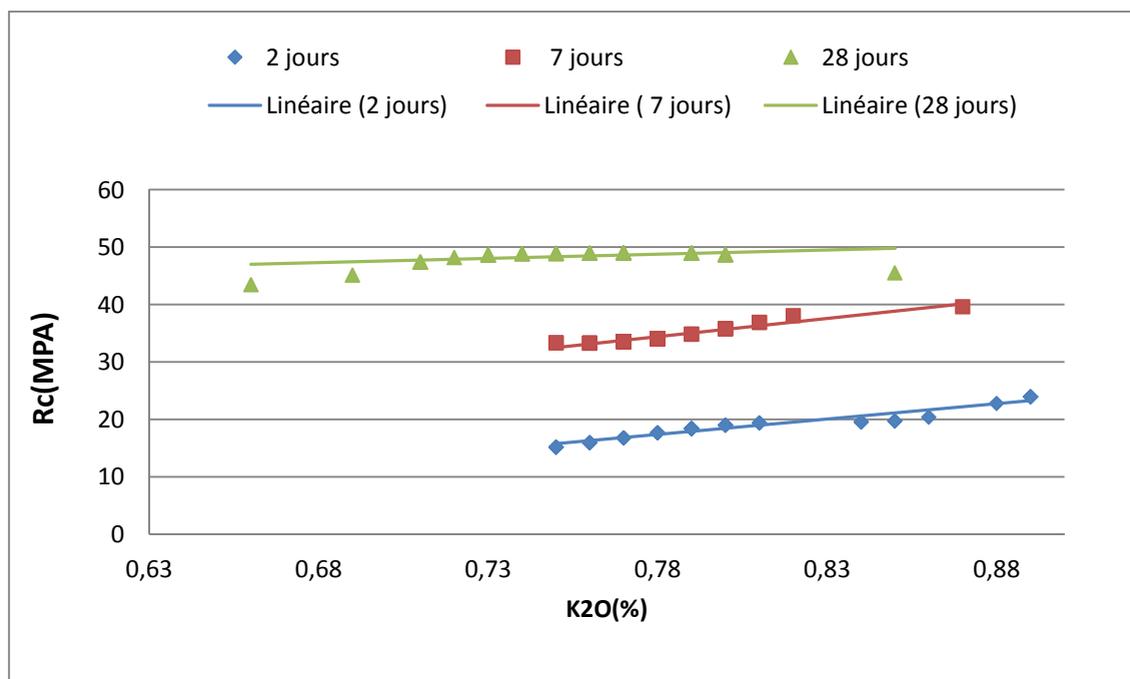
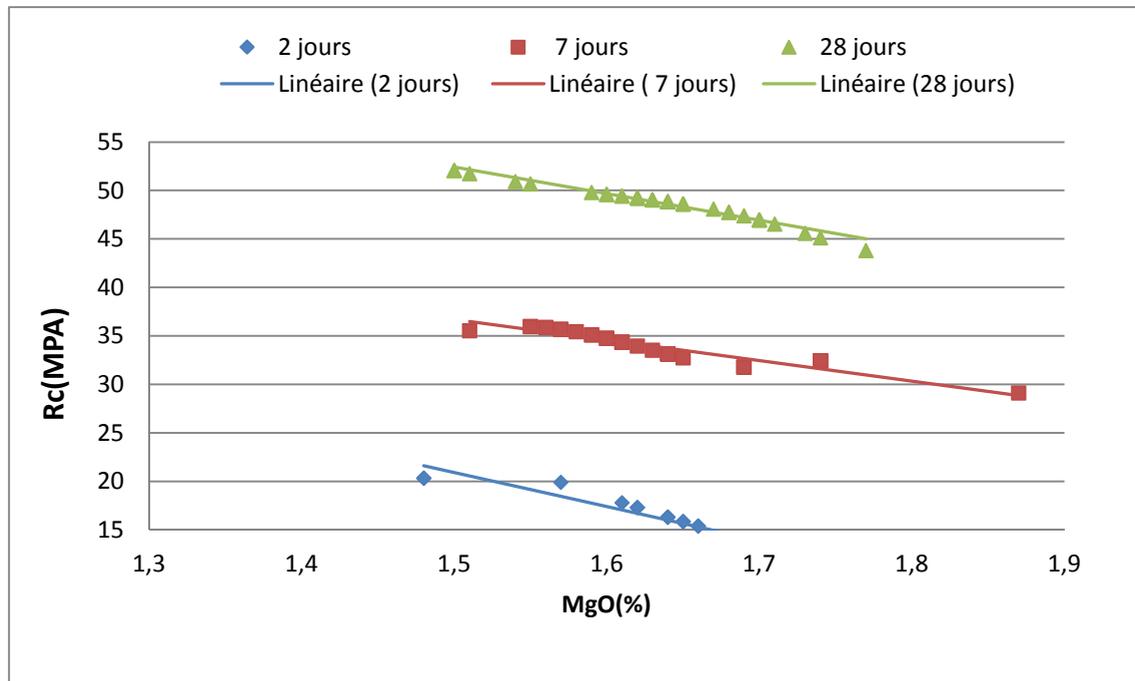


Figure (4-9) Influence des alcalis sur les résistances à la compression

Les alcalis sont favorables au développement des résistances initiales et contrarient les résistances finales.

b) Influence de MgO :



Figure(4-10): Influence de MgO sur les résistances à la compression

L' Oxyde de magnésium est un facteur défavorable pour le développement des résistances. Une augmentation de l'Oxyde de magnésium est le signe d'une dégradation de la résistance à la compression.

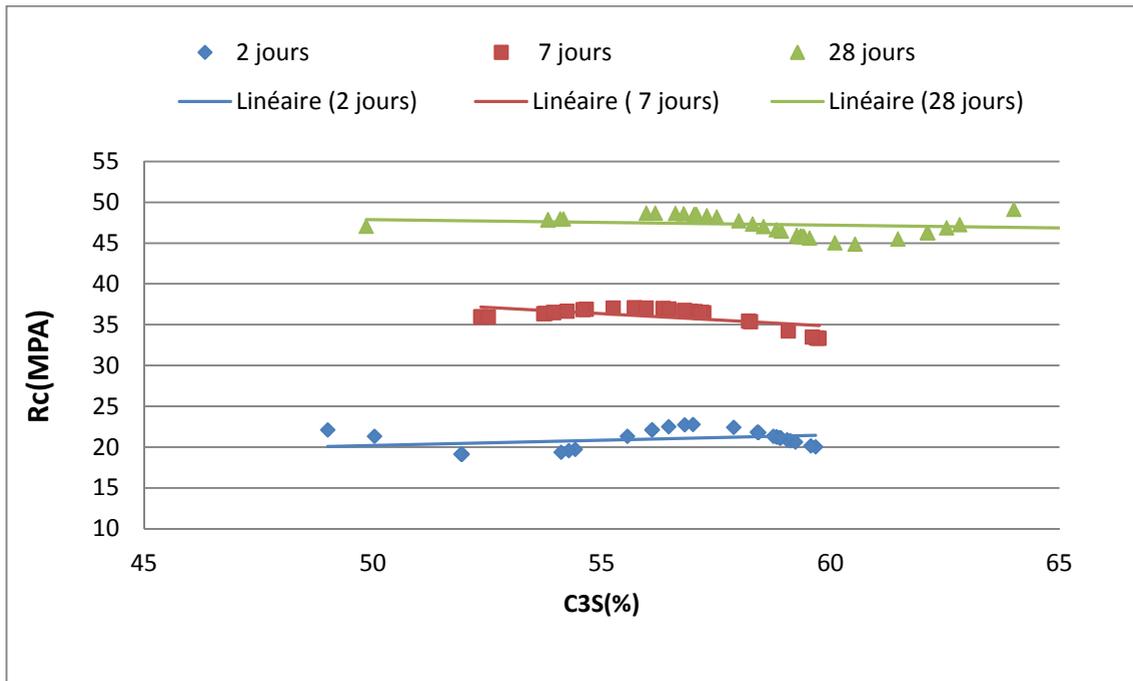
c) Influence du C_3S :

Figure (4-11) Influence du C_3S sur les résistances à la compression d'un pate du clinker

Une augmentation du C_3S est favorable au développement des résistances même au cours du temps, un ciment à haute teneur en C_3 sa réaction rapide avec l'eau donne une résistance à jeune âge élevée.

d) Influence du C₂S :

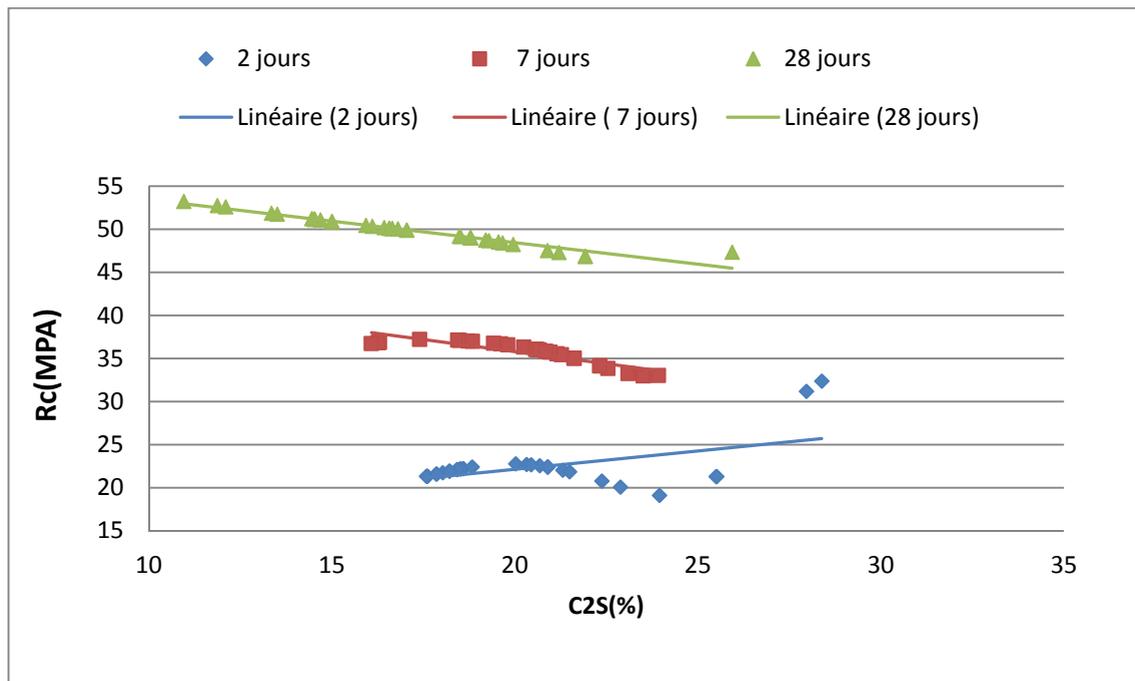


Figure (4-12) Influence du C₂S sur les résistances à la compression

Une augmentation du C₂S (ou une réduction du C₃S) est défavorable au développement initial des résistances. Par contre à 28 jours, l'augmentation du C₂S compense la diminution du C₃S.

e) Influence du C₃A :

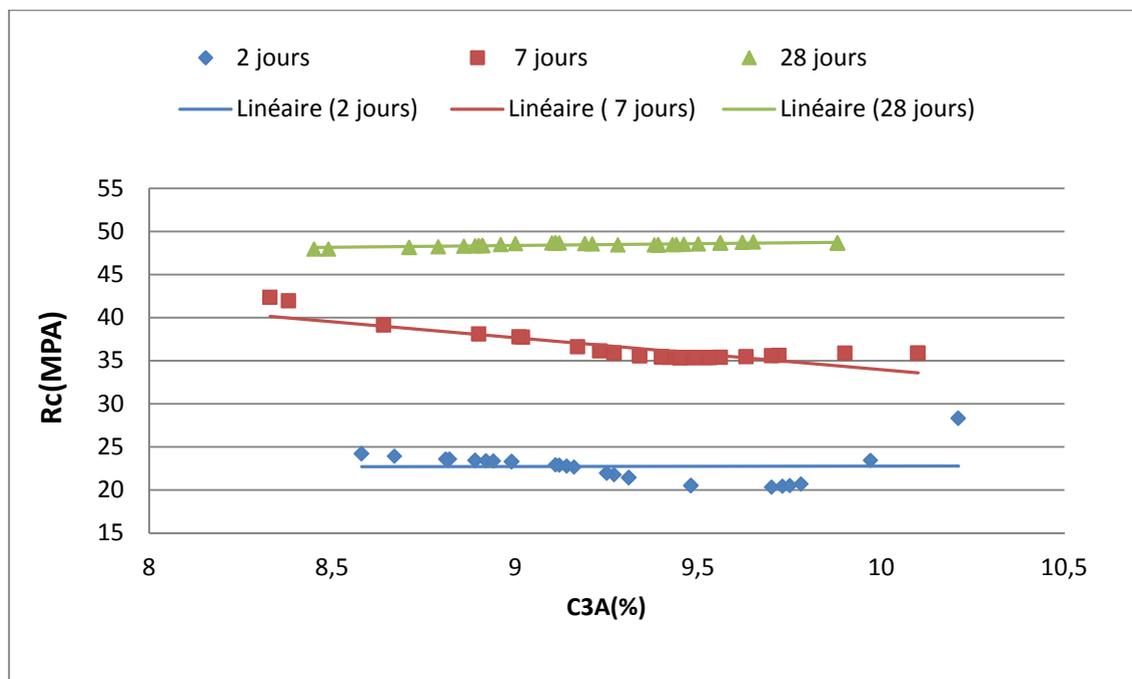


Figure (4-13) Influence du C₃A sur les résistances à la compression

Une augmentation du C_3A est favorable au développement des résistances ,Dans les conditions d'essais, l'influence du C_3A sur les résistances n'est pas mis en évidence.

Les résistances à 24 heures ne sont pas mesurées.

On développera dans le sous titre suivant le Modèle Global de l'Expansion en suivant la même méthodologie et les mêmes démarches que pour la modélisation de résistance.

4.4.3. Modèle Global de l'expansion

i. Analyse Chimique

Les données sont divisées en trois parties (70% pour l'apprentissage, 15% pour le test et 15% pour la validation).

Le tableau 4-9 récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations .

Tableau (4-9): Indicateurs utilisés – plage de variation.

Type de paramètres	Description	Plage de variation
Entrées	SiO_2 : Silice	$20.3 < SiO_2 < 22.6$
	Al_2O_3 : Alumine	$5.06 < Al_2O_3 < 5.79$
	Fe_2O_3 : Oxyde ferrique	$2.7 < Fe_2O_3 < 3.52$
	c_{a0} : la chaux	$62.8 < c_{a0} < 66.6$
	M_{g0} : Oxyde de magnésium	$1.48 < M_{g0} < 1.87$
	k_{20} : Oxyde de potassium	$0.64 < k_{20} < 0.89$
	Na_{20} : Oxyde de sodium	$0.09 < Na_{20} < 0.27$
Sorties	Expansion de la pâte de clinker	$0.63\text{mm} < \text{Exp} < 3.8 \text{ mm}$

Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère ici aussi le plus efficace et le plus performant.

L'analyse des résultats obtenus lors des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de taille RN 7-50-30-1 pour le modèle de prévision de l'expansion.

L'architecture (topologie) du réseau à rétropropagation adopté est montrée sur la Fig. 4.14:

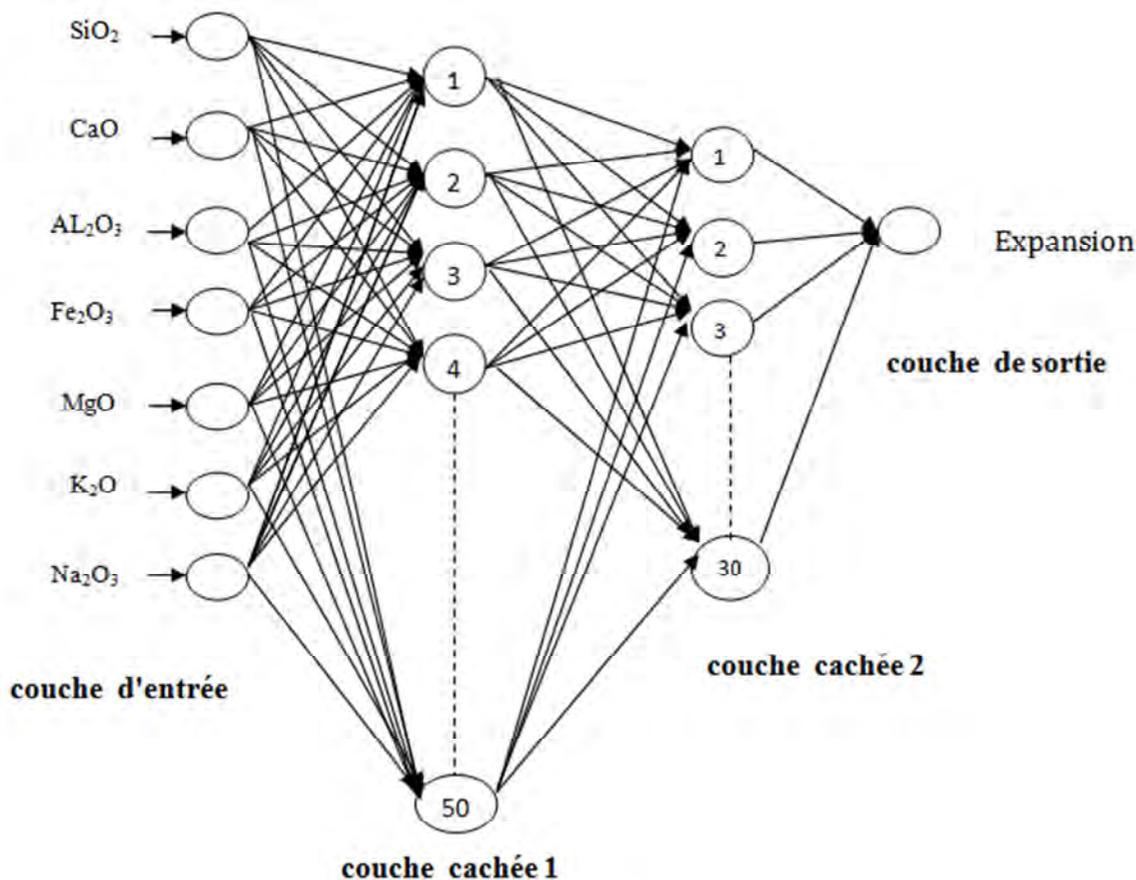
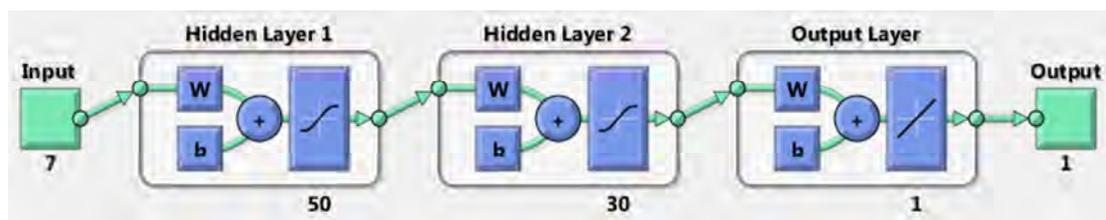


Figure (4-14) : Architecture adoptée pour l'expansion en fonction de la composition chimique de clinker.

le tableau 4-10 et la figure 4-15 ,la figure 4-16 donnent les résultats des déférents phases d'application de modèle MRNGPE (composition chimique) .

Tableau (4-10) : performance de modèle MRNGPE

Modèle RNA	Apprentissage	
	R ²	MSE
MRNGPE(7-50-30-1)	0.9326	0.0464

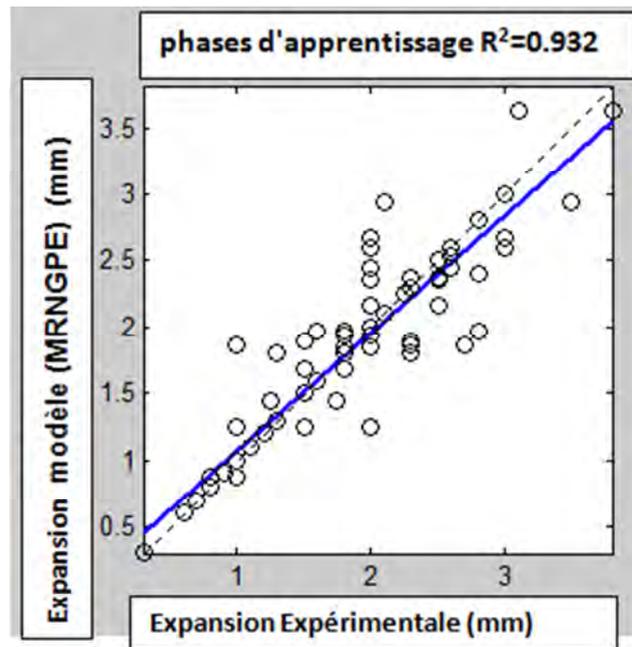


Figure (4-15) Performance du Modèle Global MRNGPE(la composition chimique).

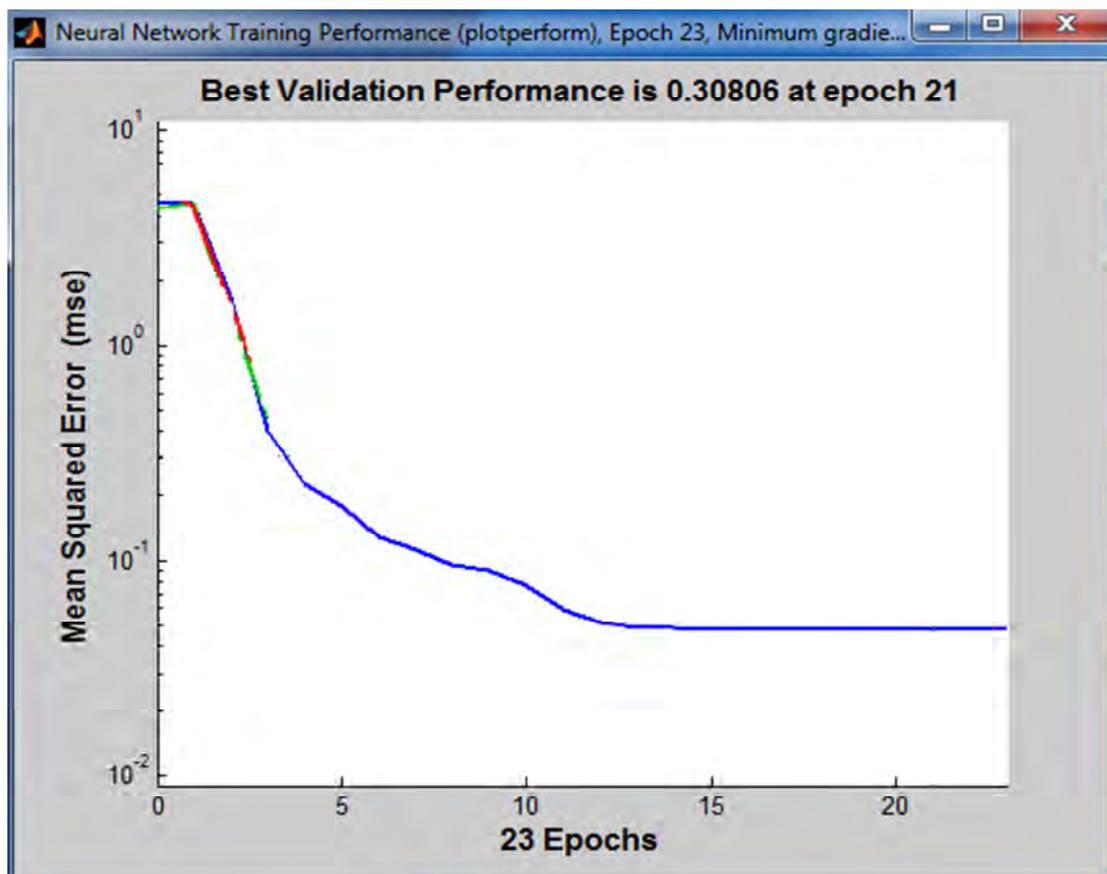


Figure (4-16) :Apprentissage du modèle MRNGPE

ii. Analyse minéralogique

Le tableau 4.11 récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations.

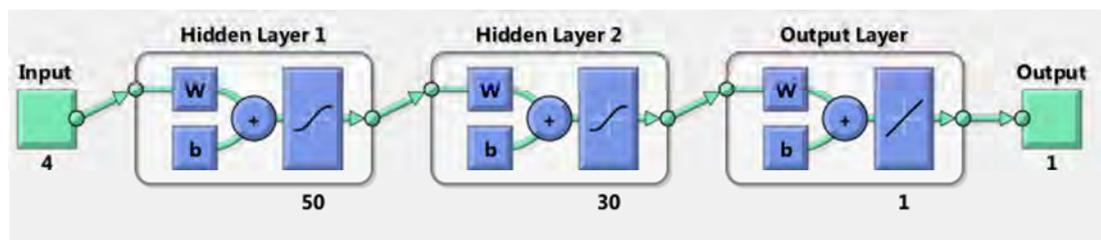
Tableau (4-11): Indicateurs utilisés – plage de variation.

Type de paramètres	Description	Plage de variation
Entrées	C ₃ S:silicate tricalcique	49 < C3S < 65.49
	C ₂ S:silicate bicalcique	10.95 < C2S < 28.38
	C ₃ A: aluminate tricalcique	7.89 < C3A < 10.21
	C ₄ AF : alumino ferrite tetracalcique	8.23 < C4AF < 11.82
Sorties	Expansion de la pâte de clinker	0.63mm < Exp < 3.8 mm

Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère ici aussi le plus efficace et le plus performant.

L'analyse des résultats obtenus lors des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de taille RN 4-50-30-1 pour le modèle de prévision de l'expansion.

L'architecture (topologie) du réseau à rétropropagation adopté est montrée sur la Fig. 4.17:



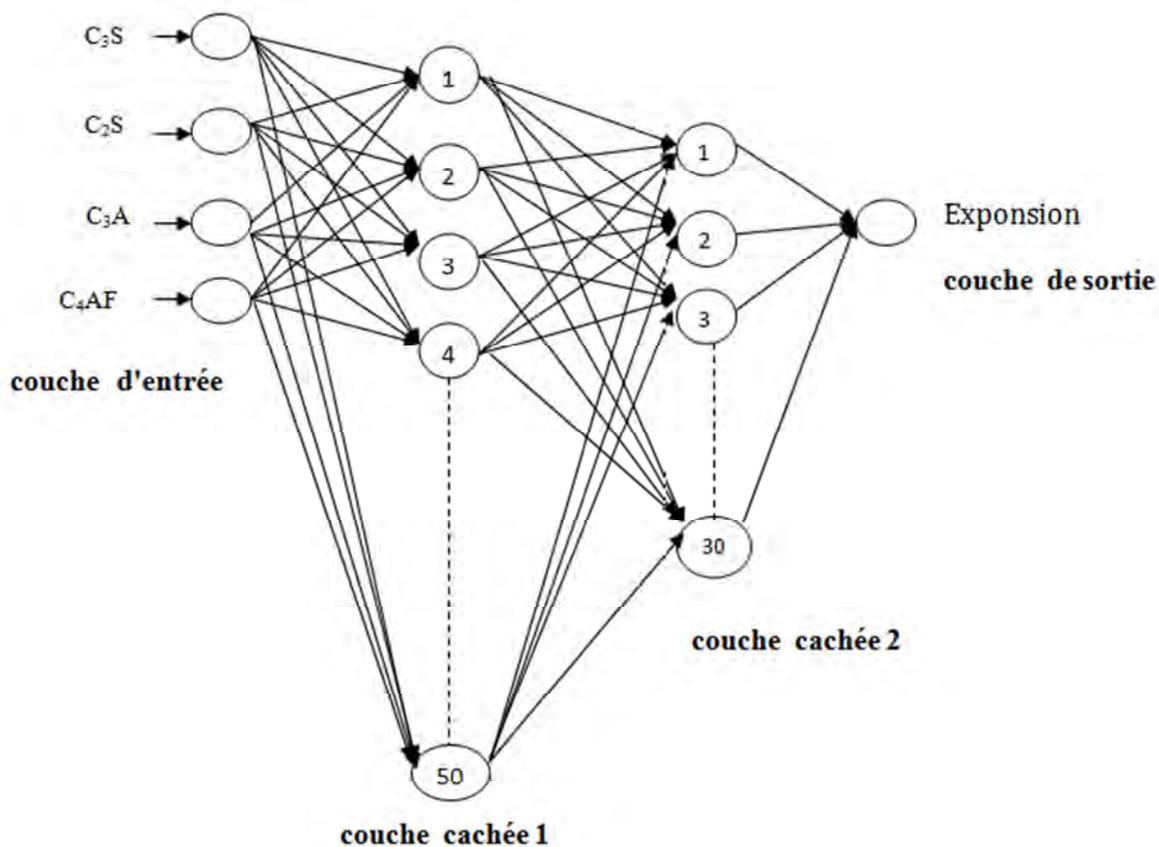


Figure (4-17) : Architecture adoptée pour l'expansion en fonction de la composition minéralogique de clinker.

le tableau 4-12 et la figure 4-18 ,la figure 4-19 donnent les résultats des différents phases d'application de modèle MRNGPE (composition minéralogique) .

Tableau (4-12) : performance de modèle MRNGPE

Modèle RNA	Apprentissage	
	R^2	MSE
MRNGPE(4-50-30-1)	0.9512	0.0332

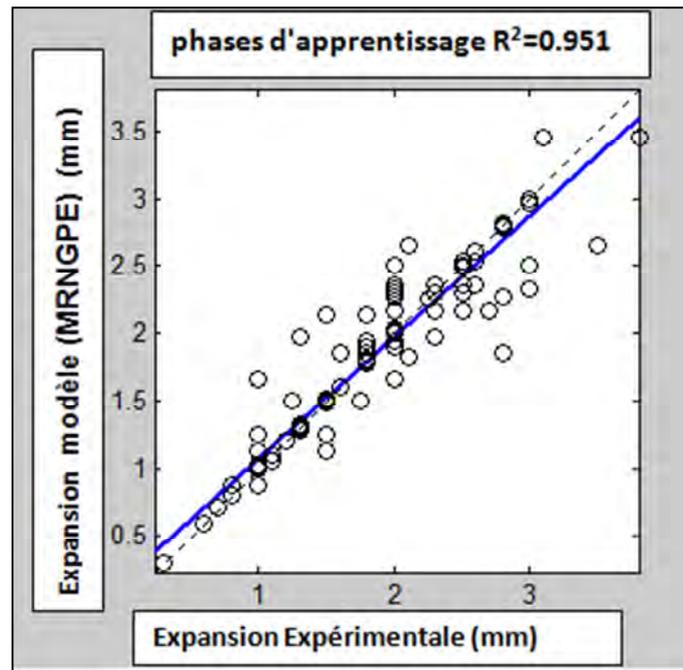


Figure (4-18) Performance du Modèle Global MRNGPE (la composition minéralogique).

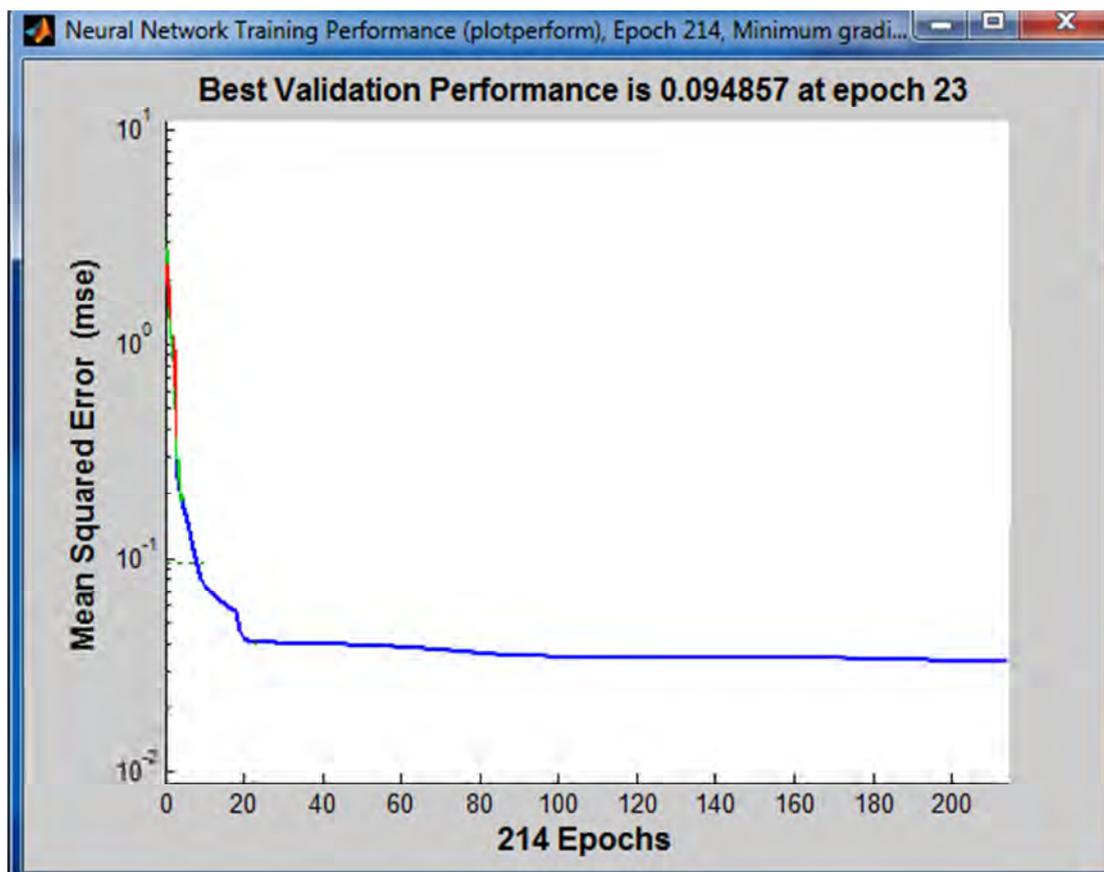


Figure (4-19) Apprentissage du modèle MRNGPE (la composition minéralogique).

iii. Etude paramétrique

Le but de cette étude paramétrique est de quantifier l'effet de certains de ces composants en utilisant le modèle de réseau de neurones (MRNGPE).

a) Influence de SiO_2 :

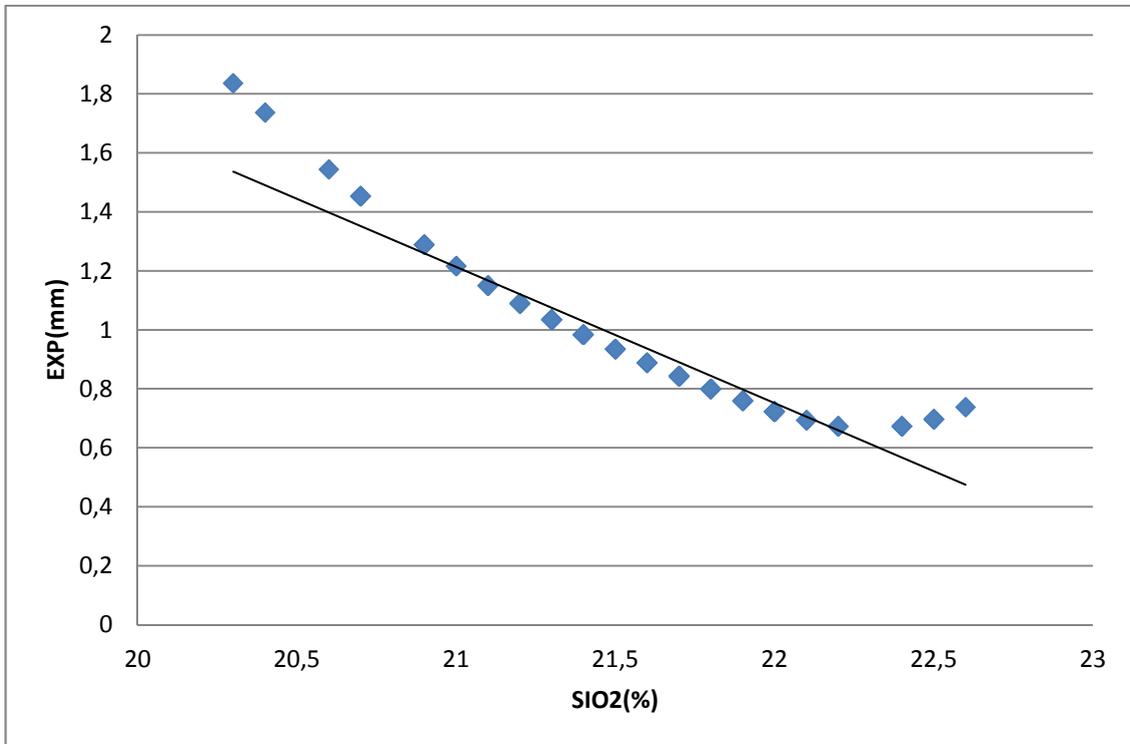
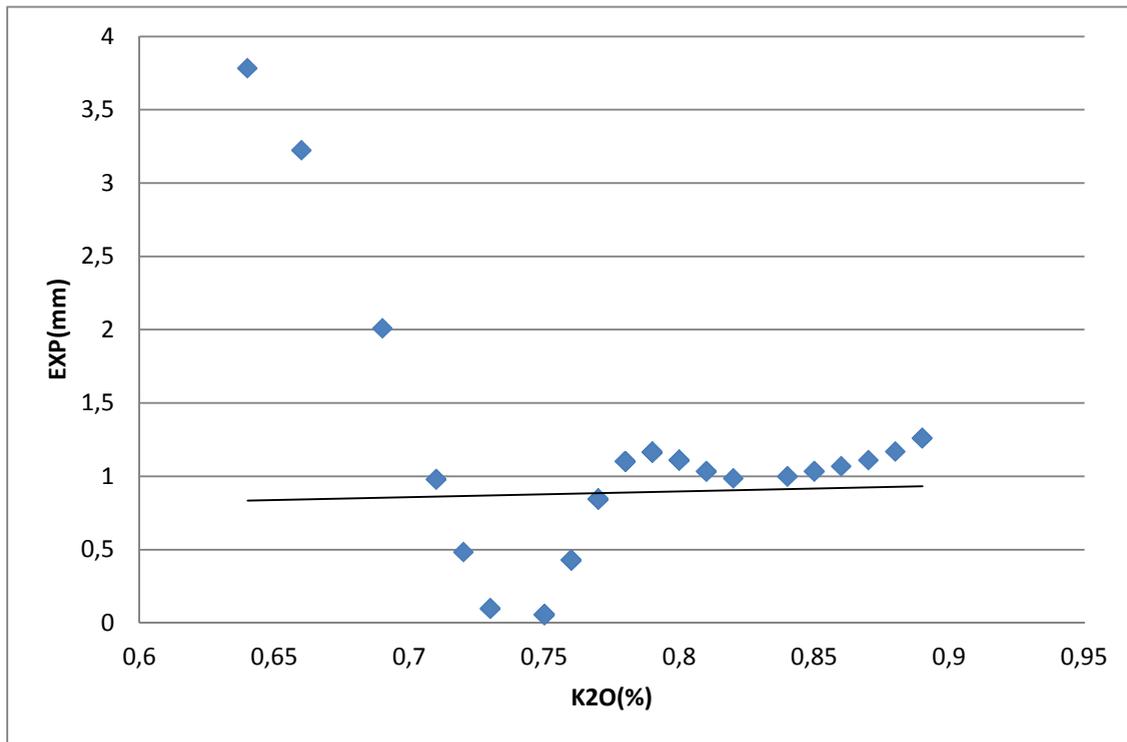


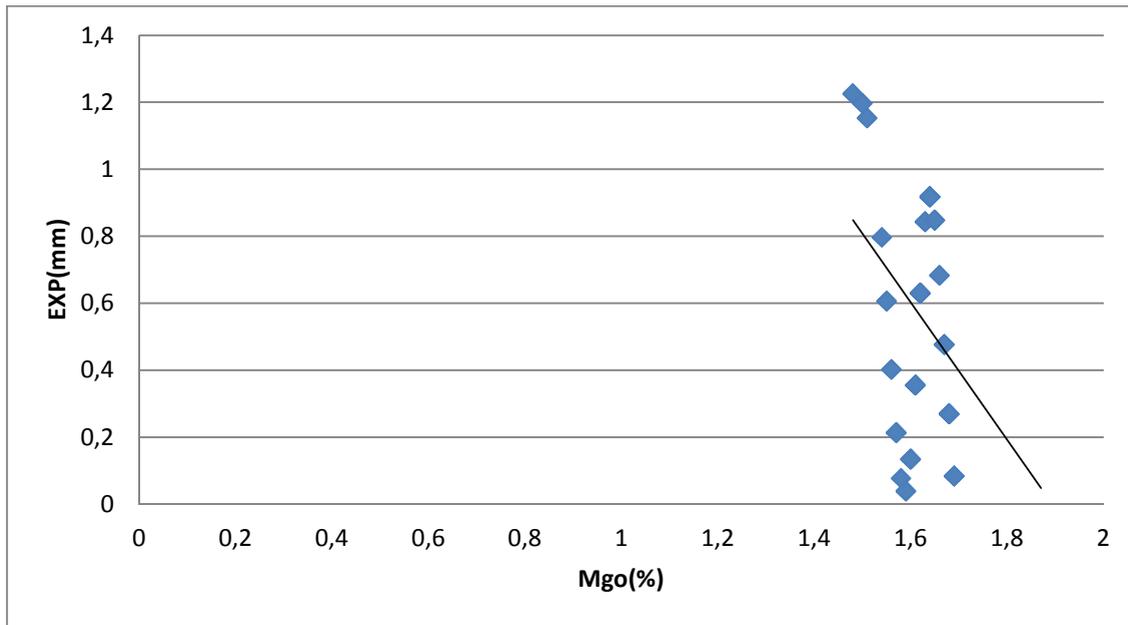
Figure (4-20) Influence de SiO_2 sur l'expansion

Le Silice est un facteur défavorable pour l'augmentation de l'expansion. Une augmentation de Silice est le signe d'une démunissions de l'expansion .

b) Influence des alcalis (K_2O)**Figure (4-21)** Influence des alcalis sur l'expansion

ce qui concerne la courbe, on constate que la variation de pourcentage de K_2O peut être déviser en 3 phases:

- ❖ phase 1 (0,65-0,75%): phase où l'expansion diminue d'une manière rapide avec l'augmentation du pourcentage de K_2O .
- ❖ phase 2 (0,75-0,80%): phase où l'expansion augmente avec l'augmentation du pourcentage de K_2O .
- ❖ phase 3 (0,8-0,9%): où la variation de l'expansion d'être stable avec l'augmentation du pourcentage du pourcentage de K_2O .

c) Influence de MgO :**Figure (4-22)** Influence de MgO sur l'expansion

L' Oxyde de magnésium est un facteur défavorable pour le développement de l'expansion. Une augmentation de l'Oxyde de magnésium est le signe d'une diminution de l'expansion.

4.4.4.Modèle Global de la consistance**i. Analyse Chimique**

Les données sont divisées en trois parties (70% pour l'apprentissage, 15% pour le test et 15% pour la validation).

Le tableau 4-13 récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations .

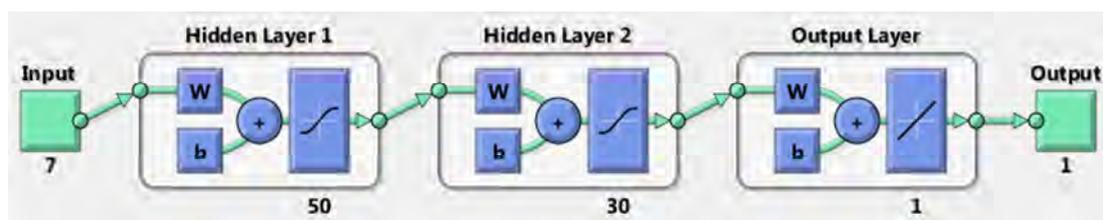
Tableau (4-13): Indicateurs utilisés – plage de variation.

Type de paramètres	Description	Plage de variation
Entrées	SiO_2 : Silice	$20.3 < SiO_2 < 22.6$
	Al_2O_3 : Alumine	$5.06 < Al_2O_3 < 5.79$
	Fe_2O_3 : Oxyde ferrique	$2.7 < Fe_2O_3 < 3.52$
	c_{aO} : la chaux	$62.8 < c_{aO} < 66.6$
	M_{gO} : Oxyde de magnésium	$1.48 < M_{gO} < 1.87$
	k_2O : Oxyde de potassium	$0.64 < k_2O < 0.89$
	Na_2O : Oxyde de sodium	$0.09 < Na_2O < 0.27$
Sorties	consistance de la pâte de clinker	$24 \% < Con < 30 \%$

Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère ici aussi le plus efficace et le plus performant.

L'analyse des résultats obtenus lors des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de taille RN 7-50-30-1 pour le modèle de prévision de la consistance .

L'architecture (topologie) du réseau à rétropropagation adopté est montrée sur la Fig. 4.23 :



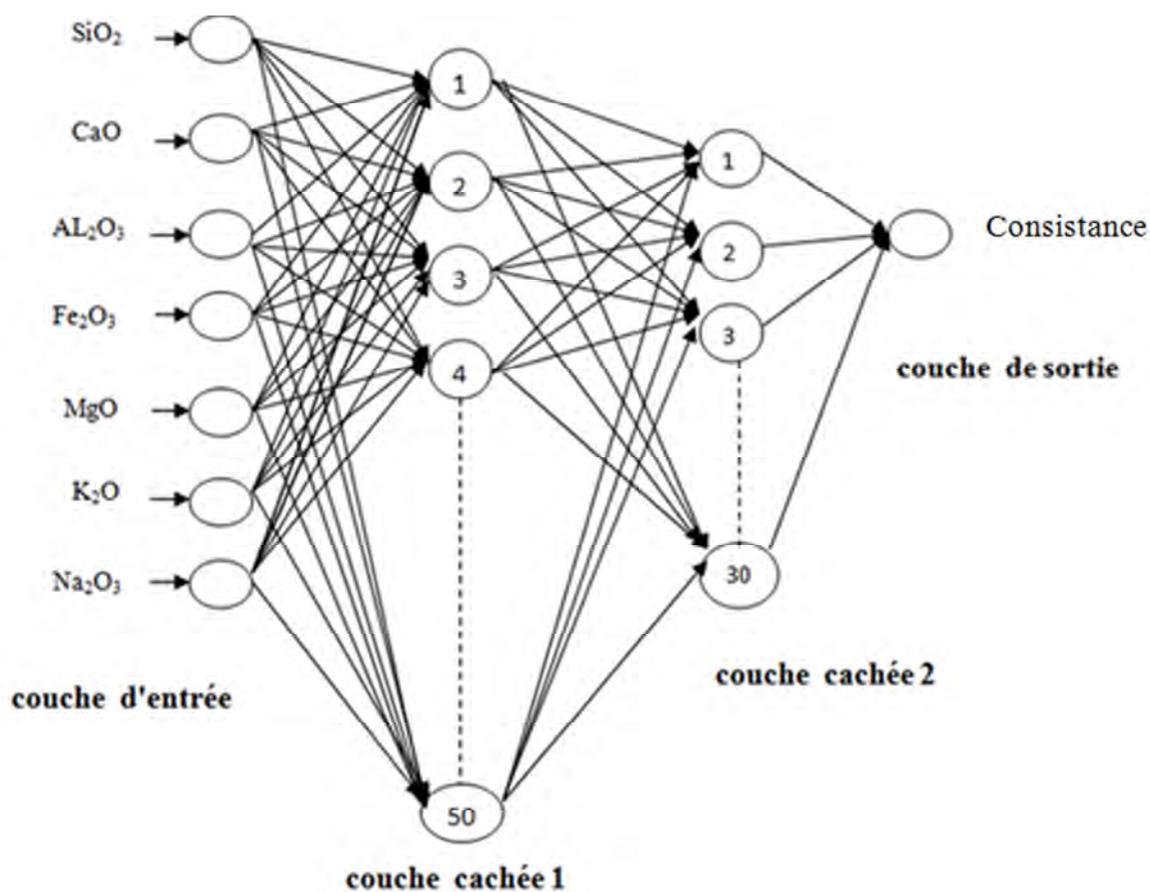


Figure (4-23) Architecture adoptée pour la consistance en fonction de la composition chimique de clinker.

le tableau 4-14 et la figure 4-24 ,la figure 4-25 donnent les résultats des différents phases d'application de modèle MRNGPC (composition chimique) .

Tableau(4-14): performance de modèle MRNGPC

Modèle RNA	Apprentissage	
	R ²	MSE
MRNGPC(7-50-30-1)	0.978	0.0444

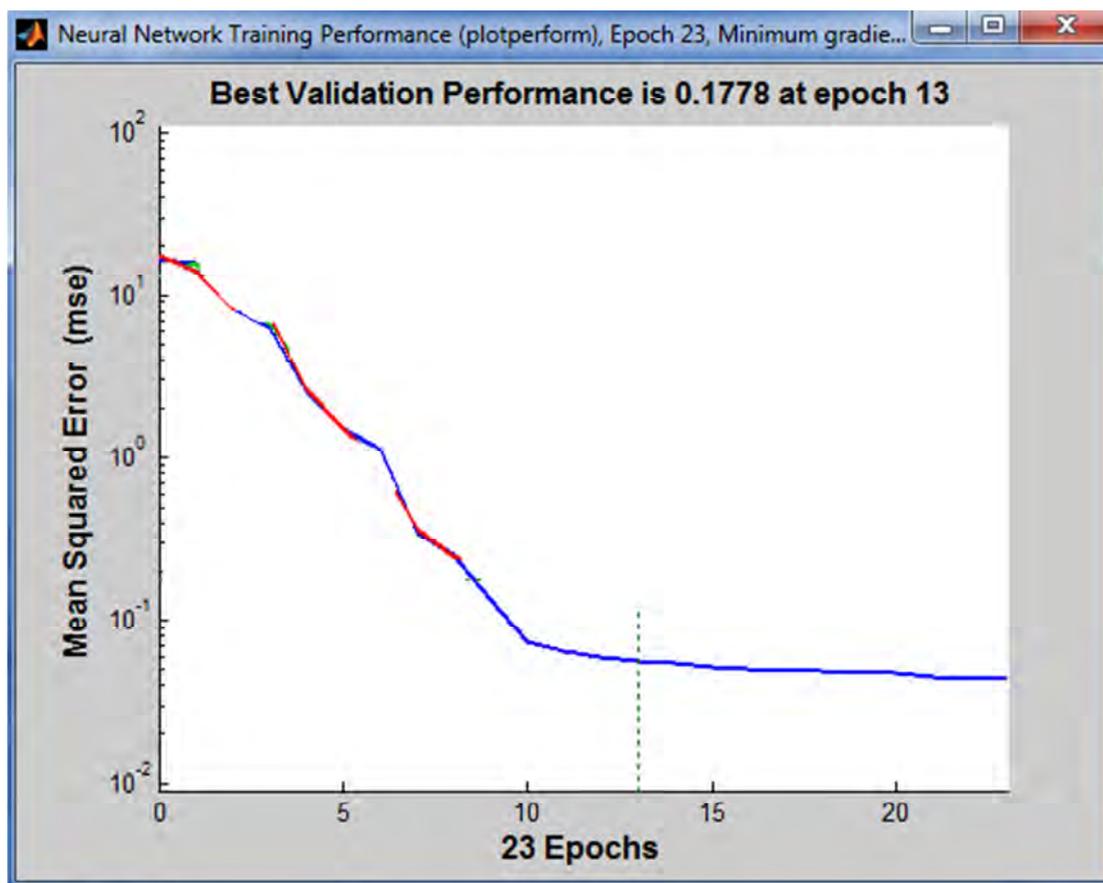


Figure (4-24) Apprentissage du modèle MRNGPC (composition chimique).

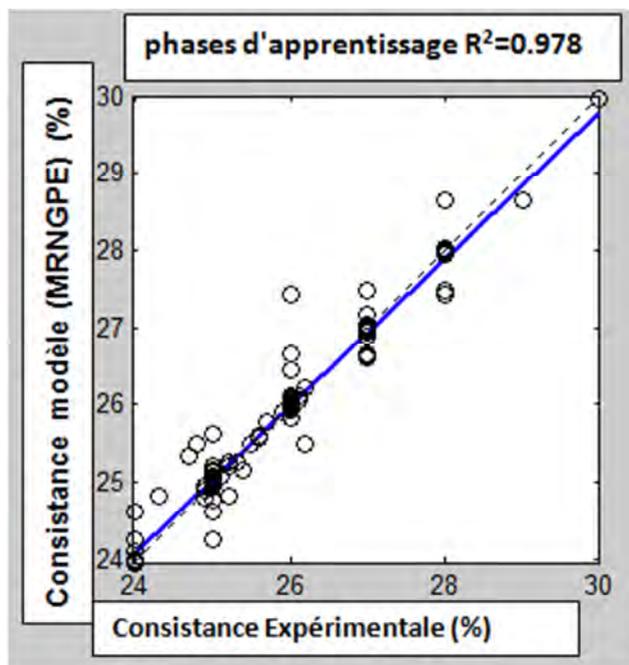


Figure (4-25) Performance du Modèle Global MRNGPC(composition chimique).

ii. Analyse minéralogique

Le tableau 4.15 récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations.

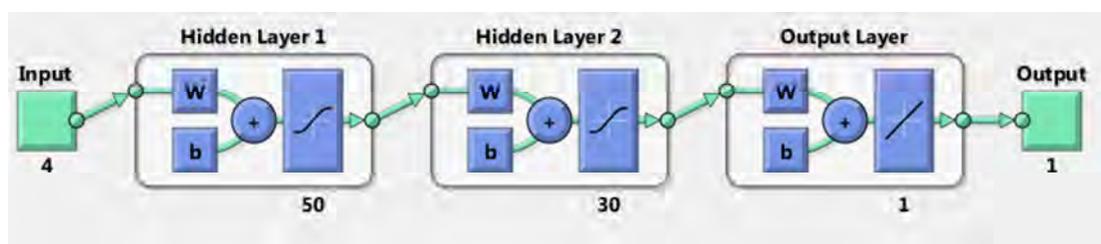
Tableau (4-15): Indicateurs utilisés – plage de variation.

Type de paramètres	Description	Plage de variation
Entrées	C ₃ S:silicate tricalcique	49 < C3S < 65.49
	C ₂ S:silicate bicalcique	10.95 < C2S < 28.38
	C ₃ A: aluminate tricalcique	7.89 < C3A < 10.21
	C ₄ AF : alumino ferrite tetracalcique	8.23 < C4AF < 11.82
Sorties	Consistance de la pâte de clinker	0.63mm < Con < 3.8 mm

Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère ici aussi le plus efficace et le plus performant.

L'analyse des résultats obtenus lors des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de taille RN 4-50-30-1 pour le modèle de prévision de l'expansion.

L'architecture (topologie) du réseau à rétropropagation adopté est montrée sur la Fig (4-26).



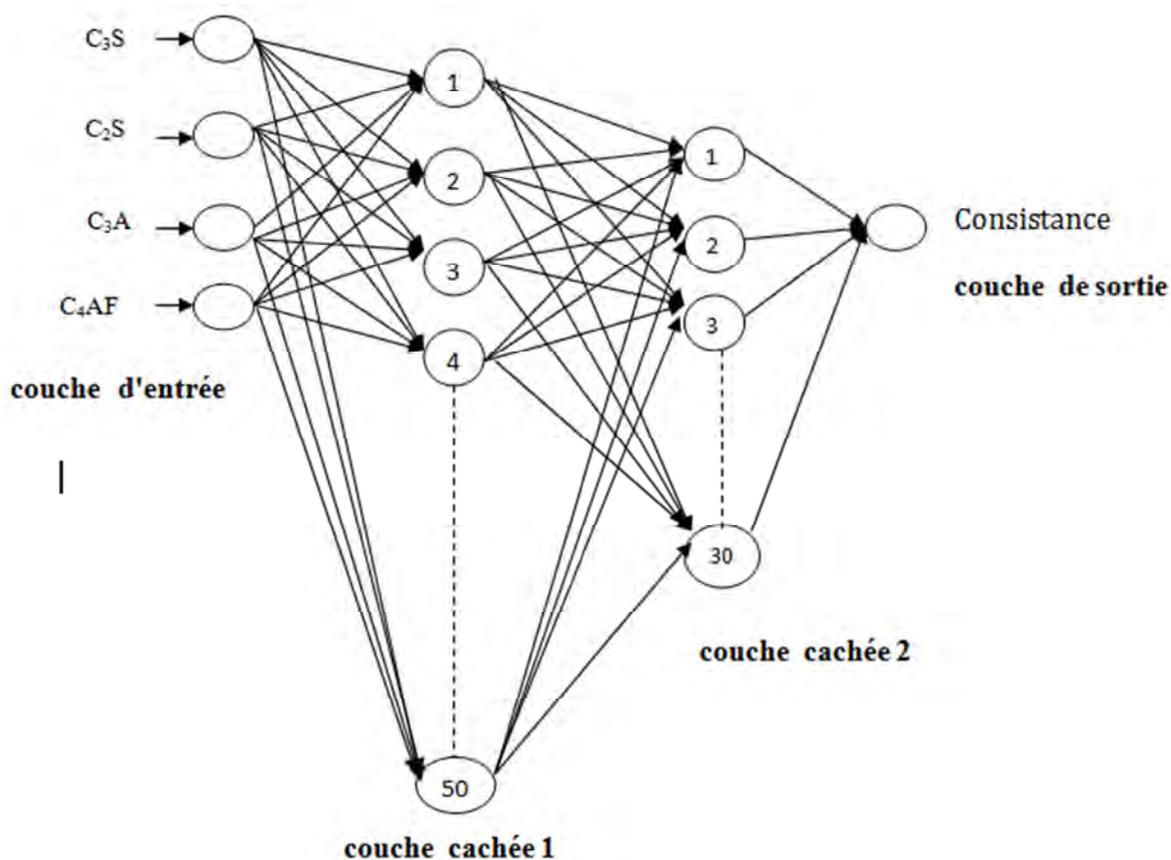


Figure (4-26) : Architecture adoptée pour la consistance en fonction de la composition minéralogique de clinker.

le tableau 4-16 et la figure 4-27 ,la figure 4-28 donnent les résultats des différents phases d'application de modèle MRNGPC (composition minéralogique) .

Tableau (4-16) : performance de modèle MRNGPC

Modèle RNA	Apprentissage	
	R^2	MSE
MRNGPC(4-50-30-1)	0.966	0.0444

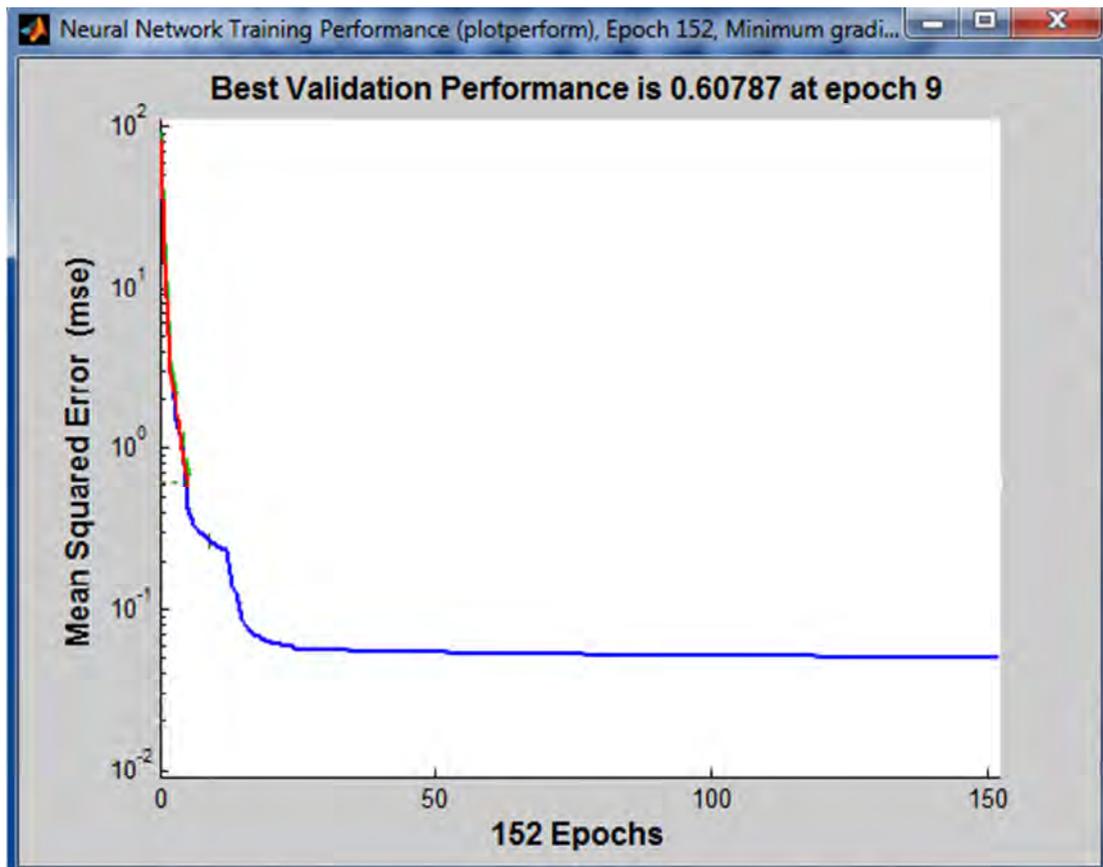


Figure (4-27) Apprentissage du modèle MRNGPC (composition chimique).

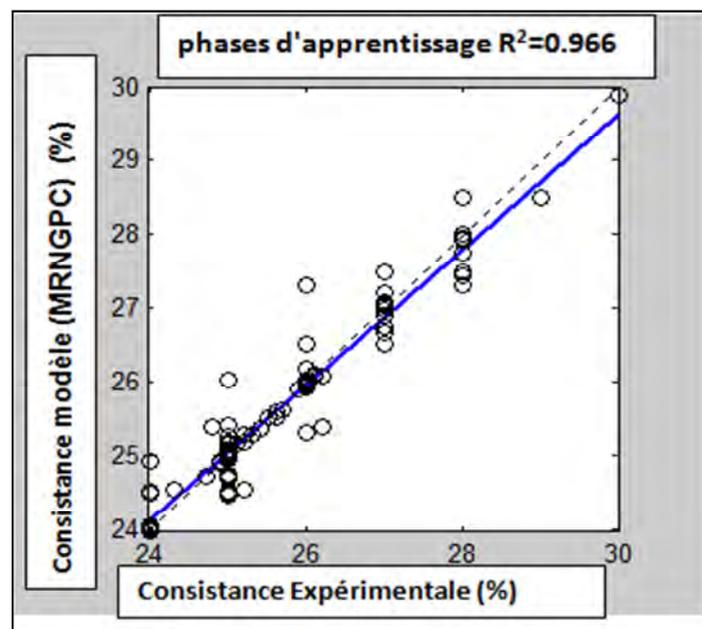


Figure (4-28) Performance du Modèle Global MRNGPC(composition chimique).

iii. Etude paramétrique

Le but de cette étude paramétrique est de quantifier l'effet de certains de ces composantes en utilisant le modèle de réseau de neurones (MRNGPC).

a) Influence de SiO_2 :

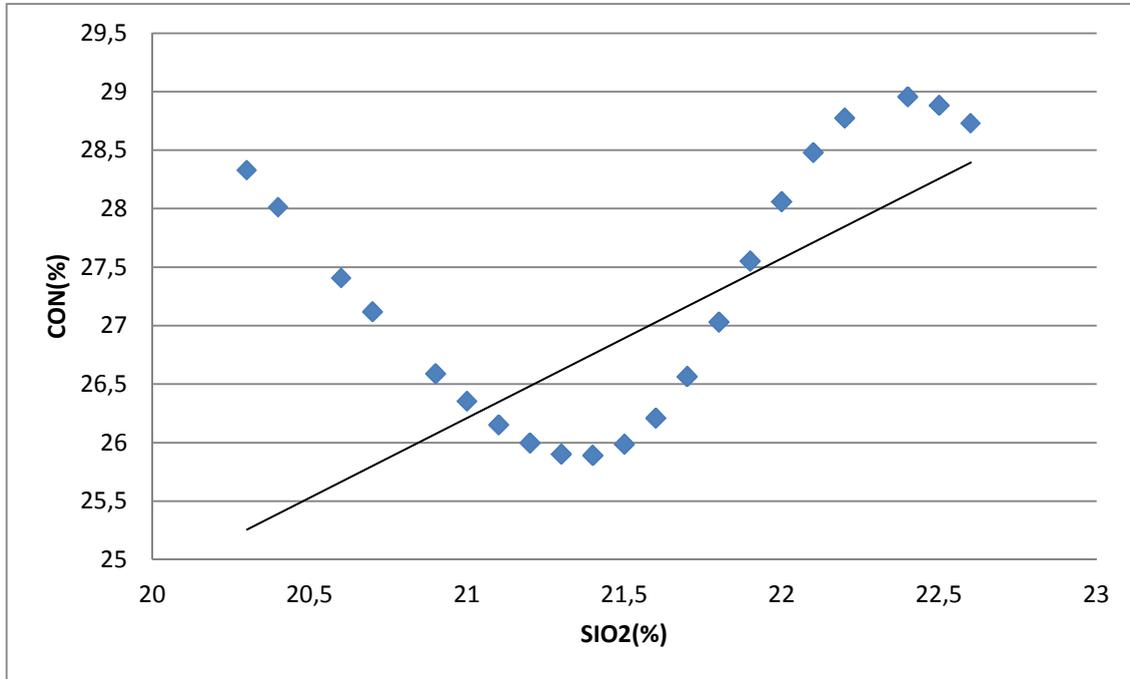


Figure (4-29) Influence de SiO_2 sur la consistance

on constate d'après la courbe que la consistance diminue avec l'augmentation du pourcentage de silice entre [20.3-21.4 %], au-delà de cette intervalle la consistance augmente .

Donc le silice est un facteur favorable pour l'augmentation de la consistance. Une augmentation de SiO_2 à certain limites est le signe d'une augmentation la consistance .

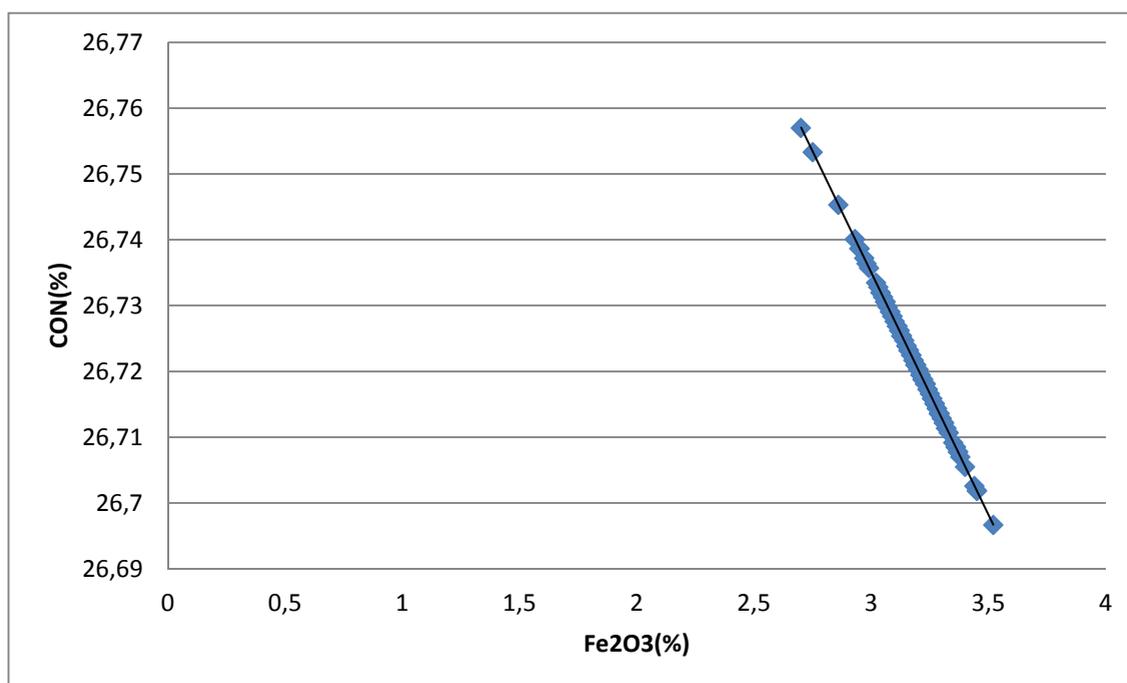
b) Influence de Fe_2O_3 :

Figure (4-30) Influence de Fe_2O_3 sur la consistance

L'Oxyde ferrique est un facteur défavorable pour le développement de la consistance. Une augmentation de L'Oxyde ferrique est le signe d'une diminution de la consistance .

c) Influence des alcalis Na_2O :

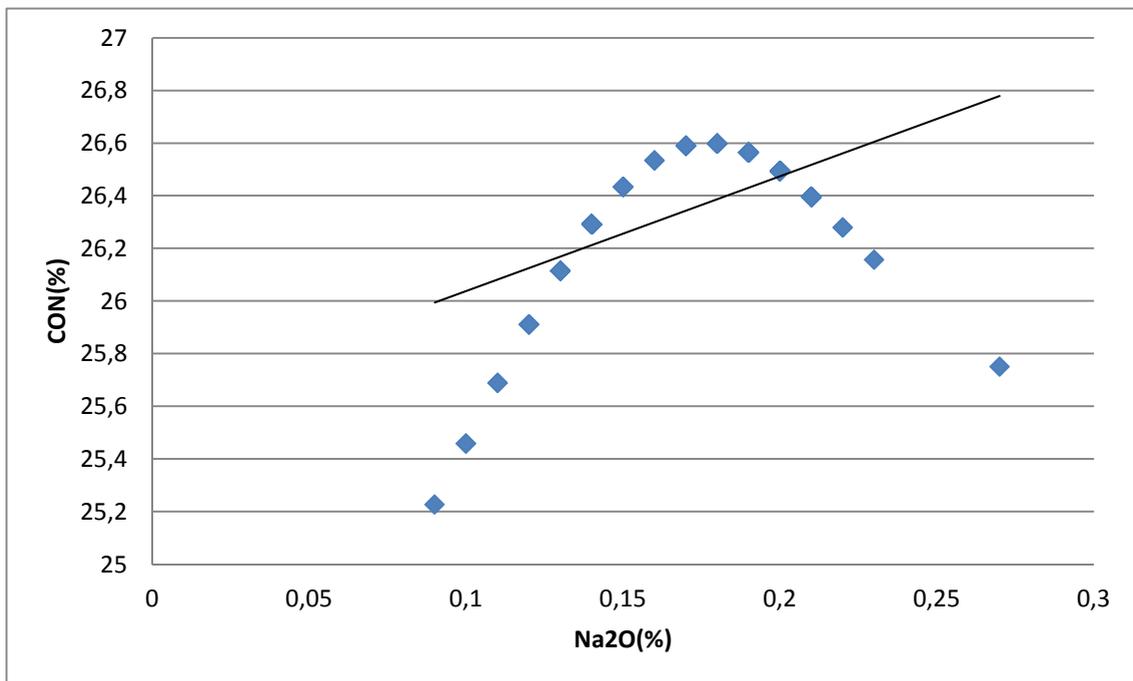


Figure (4-31) Influence des alcalis sur la consistance.

L'Oxyde de sodium est un facteur favorable pour l'augmentation de la consistance. Une augmentation de Na_2O est le signe d'une augmentation la consistance.

d) Influence des alcalis (K_2O) :

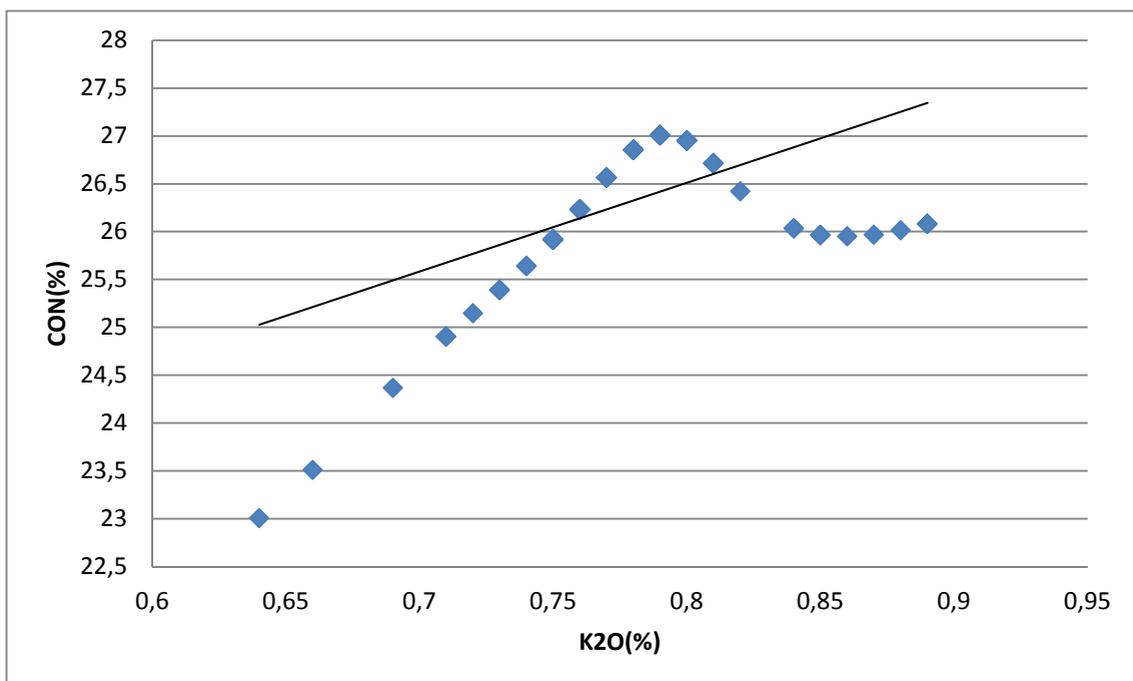


Figure (4-32): Influence des alcalis sur la consistance

L'Oxyde de potassium est un facteur favorable pour l'augmentation de la consistance. Une augmentation de K_2O est le signe d'une augmentation la consistance entre [0.65-0.80%], au-delà c'est l'inverse .

4.5. Conclusion

Les alcalins sont très importants par leur influence sur les propriétés du clinker, le développement de la résistance à la compression est probablement liée a l'augmentation de la teneur en silice.

Cette étude a permis d'aboutir à un modèle décrivant l'évolution des variations dimensionnelles des composition chimique et minéralogique de clinker au résistance mécanique, l'expansion et la consistance. Les modèles MRNGPR et MRNGPE et MRNGPC présentent l'avantage de prendre en considération tous les paramètres influant sur le résistance mécanique, l'expansion et la consistance.

5. MODELES RESEAUX DE NEURONES SUR LES BETONS

5.1 Le béton hydraulique

Le béton est un matériaux obtenues en solidarissant, par une pâte liant de ciment, une squelette granulaire composés d'un ou plusieurs sables, et d'un ou plusieurs graviers. Comme les liants à des propriétés hydrauliques, ces bétons sont aussi appelés bétons hydrauliques[31].

L'impotence de la qualité des liants et ces compositions tiennent à leurs relations plus ou moins directes avec les propriétés du béton. Ces derniers dépend évidemment de nombreuses autres paramètres (rapport E/C, aires occlus, caractéristiques des autres constituants des sables et graviers ...), ce qui oblige à raisonner ici "toutes choses égales par ailleurs", c'est-à-dire les autres constituants du béton étant les même et les proportions restent inchangées.

L'étude de la composition du béton consiste à définit le mélange optimale des différents granulats dont on dispose, aussi le dosage en ciment et en eau, afin de réalisé un béton dont les qualités soit celles recherchées pour la construction de l'ouvrage en cause.

Les objectifs requise d'une telle composition d'un béton sont possible, mes notre but est non seulement la formulation mais aussi de vérifier l'influence de type de ciment utilisé sur les propriétés d'un béton données par cette formulation.

L'objectif de base de ce chapitre est d'utilisé le laitier comme liant hydraulique dans un béton ordinaire. Pour cela on à établi un modèle de réseaux de neurone basé sur des résultats expérimentaux des bétons comprend des différents pourcentages du laitier.

Les résultats expérimentaux dans cette partie sont celles qui utilisées dans le travaille du groupe de professeur GUETTECHE M.N

5.2 les constituants utilisés

le ciment utilisé dans cette étude à été élaboré au laboratoire des matériaux . on a utilisé le clinker prévenant de la ligne (1) de la cimenterie de Hadjar Soud et du laitier granulé de El Hadjar , ensuit on a procédé à leur broyage à l'aide de l'appareil de Los Angeles jusqu'a l'obtention d'une finesse semblable à celle des ciment ordinaires .les deux matériaux ont été broyés séparément puis homogénéisés en ajoutant une quantité de 6 % de gypse .

cette opération nous a permet d'obtenir un ciment au laboratoire en faisant varier les taux du clinker et du laitier.

les granulats concassés proviennent de la carrière "Hedna" (la commune de El KHROUB)

ce gisement est constitué essentiellement de calcaire à grains fins lités parfois alternant avec des couches marneuses et marno- calcaire dolomitiques et calcaire .Il y a une prédominance de calcaire dolomitiques et calcaire cristallines .La partie supérieure est formée de calcaire à grains fins de couleur blanche à grisâtre et sont dispose en banc.

❖ Analyse granulométrique

les classe granulaire et les résultats de l'analyse granulométrique conformément à la norme NFP18-560;NA2607. Aussi les caractéristiques physico-chimiques

5.3 Etude expérimentale de la composition du béton et méthodologie de composition

Pour calculer la composition de notre béton nous avons choisi la méthode simple est pratique. Elle permet de définir d' une façon simple et rapide une formule de composition à peu près adopté au béton étudié ,mais que seuls quelques gâchées d'essais et la confection d'éprouvette permettant d'ajuster au mieux la composition à adoptée définitivement en fonction des qualités souhaitées et des matériaux effectivement utilisés .

A-Donnés de base

*Consistance moyenne désirées 7 à 9 affaissement au cône d'ABRAMS.

*Dimension maximale des granulats $D=25\text{mm}$

B-Dosage en ciment

On adopte le dosage en ciment imposé et l'on corrige en conséquence le dosage des granulats $C=350\text{ kg/m}^3$

C-Dosage en eau

La relation entre la consistance mesurée au cône d'ABRAMS et le rapport C/E pour différent dosage en ciment donnent une consistance désirés de 8cm et un dosage $C=350\text{kg/m}^3$ d'ou le rapport $C/E=1.78$, donc $E=196$ litres

D-Dosage en granulats

Une fois le trace granulaire tracée, nous avons trouvées les pourcentage suivant :

sable : 0/3 : 33%

sable : 3/8 : 11%

gravier : 8/15 : 21%

gravier : 15/25 : 35%

Le volume absolu des granulats $V = 684$ litre , le dosage :

Ciment : 350 kg/m³

Eau : 196 l/m³

Sable 0/3 : 225.72 l/m³

Sable 3/8 : 75.24 l/m³

Gravier 8/15 : 143.64 l/m³

Gravier 15/25 : 239.4 l/m³

5.4 Les résultats expérimentaux

les différents résultats obtenus à partir de quelques essais sur béton ont pour but de montrer l'influence d'un ciment à base de laitier activé sur les résistances en compression réalisées sur cube (7 x 7 x 7 cm³) ou en traction par flexion (7 x 7 x 28 cm³) effectuées sur béton durci.

Nous utilisons ces résultats expérimentaux pour l'apprentissage de programme RNA.

les résultats des résistances en compression et traction par flexion sont présentés dans le tableau (5-1).

tableau (5-1) : résultats des résistances en compression et traction des différents pourcentages de laitier.

N	Clinker %	Laitier %	Gypse %	jour	Resistance à la compression (Mpa)	Resistance à la traction (Mpa)
1	94	0	6	2	15,3	1,4
2	64	30	6	2	10,8	0,7
3	47	47	6	2	12,3	0,5
4	19	75	6	2	4	0,1
5	94	0	6	28	45	3,7
6	64	30	6	28	30	3,1
7	47	47	6	28	22,5	2,4
8	19	75	6	28	19,4	2
9	94	0	6	90	57	4,4
10	64	30	6	90	52	3,7
11	47	47	6	90	49	2,6
12	19	75	6	90	38,8	2,4

5.5 Elaboration des modèles RNA

5.5.1 Introduction

Ce sous-chapitre comporte la présentation des deux groupes de modèles ,le 1^{er} contient le modèle de résistance à la compression du béton en fonction de la variation du pourcentage du laitier dans le ciment et la 2^{ème} contient le modèle de résistance à la traction du béton en fonction de la variation du pourcentage du laitier.

Un ensemble de données provenant d'études expérimentales ont été utilisés pour les réseaux de neurones artificiels. Les données sont utilisés pour la formation du réseau et sont représenté à L'ANNEXE.

5.5.2 Modèle Global du Résistance a la compression

Le tableau (5-2) récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations.

Tableau (5-2): Indicateurs utilisés – plage de variation.

Type de paramètres	Description	Plage de variation
Entrées	Clinker	19% < Clinker < 94%
	Laitier	0% < Laitier < 75%
	Gypse	Gypse=6%
	J : les jours	2 j , 28 j , 90 j
Sorties	Résistance à la compression Du béton	4 MPa < R _c < 57 MPa

Dans un premier temps, on procède à la détermination du type d'algorithme d'apprentissage à utiliser dans notre étude. Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère le plus efficace et le plus fiable.

Une fois l'algorithme d'apprentissage choisi, on procède à la détermination du nombre de couches cachées et de neurones par couche cachée. Ceci a été effectué en faisant des

simulations numériques avec différentes architectures du réseau (en faisant varier le nombre de neurones de chaque couche cachée).

l'analyse des résultats obtenus à partir des phases d'apprentissage, conduit à l'utilisation d'un réseau de taille RN 4-5-3-1 pour le modèle de prévision du résistance à la compression du béton.

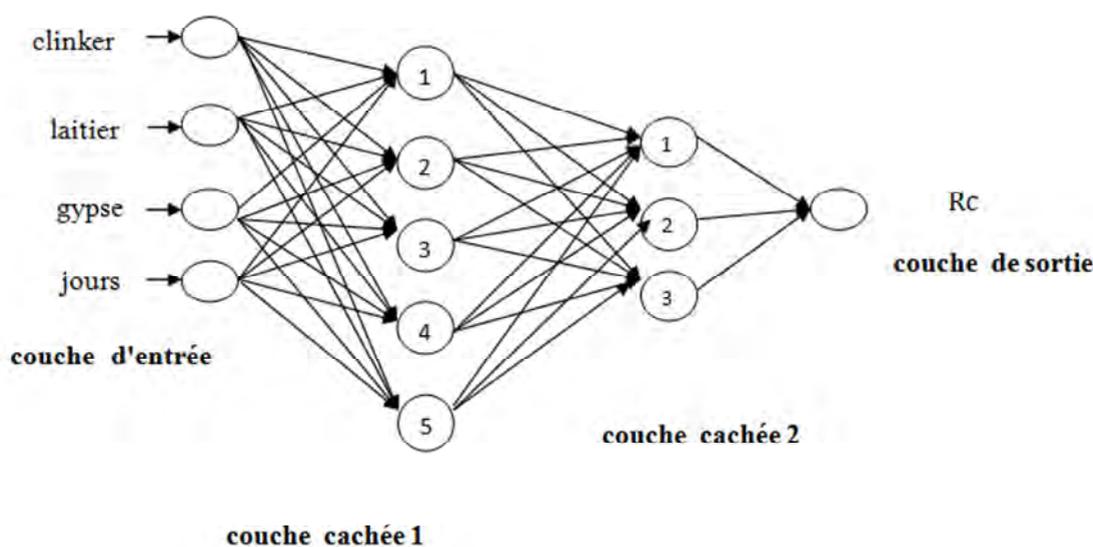
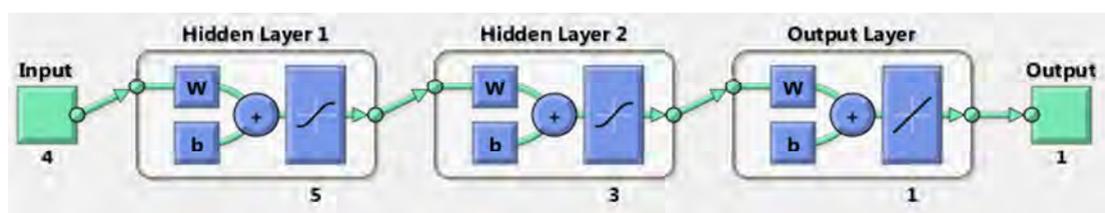


Figure (5-1) Architecture adoptée pour la Résistance à la compression du béton en fonction du pourcentage de laitier .

L'apprentissage a été basé sur l'algorithme de Levenberg-Marquardt à Rétropropagation avec les fonctions de transfert standard Tansigmoïde et Purelinéaire correspondant respectivement aux couches cachées et de sortie, avec un taux d'apprentissage de 0.01 .

le tableau (5-3) et la figure (5-2) ,la figure (5-3) donnent les résultats des différents phases d'application de modèle MRNGPR .

Tableau (5-3) : performance de modèle MRNGPR_c

Modèle RNA	Apprentissage	
	R2	MSE
MRNGPR _c (4-5-3-1)	0.999	1e-15

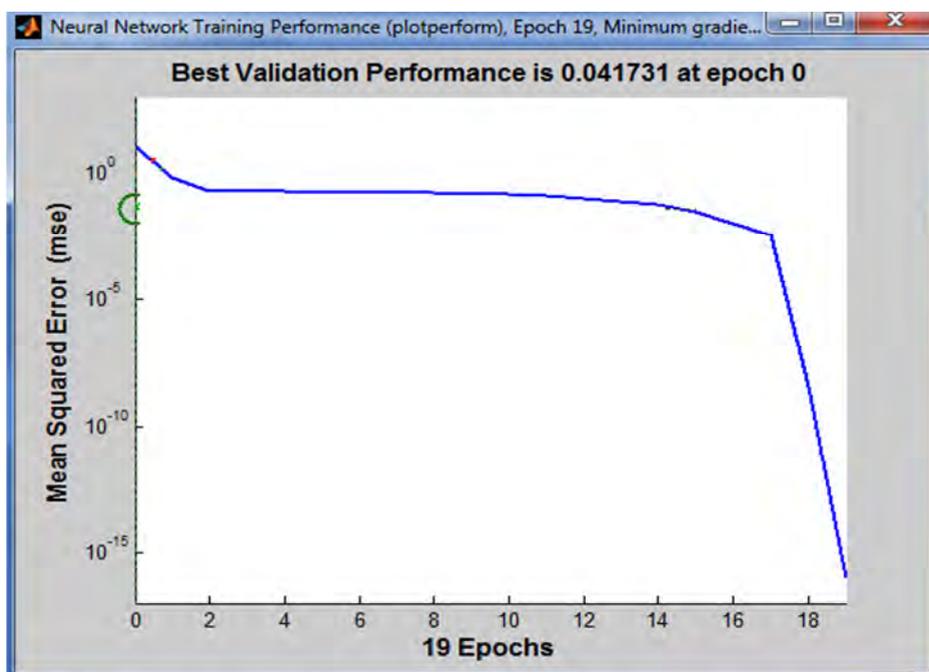
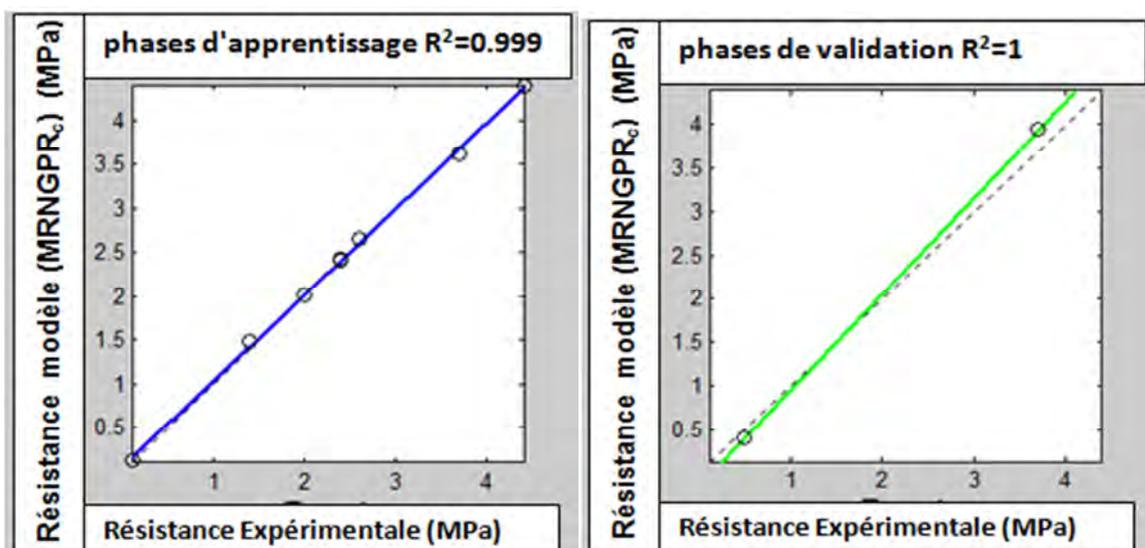


Figure (5-2) Apprentissage du modèle MRNGPR_c



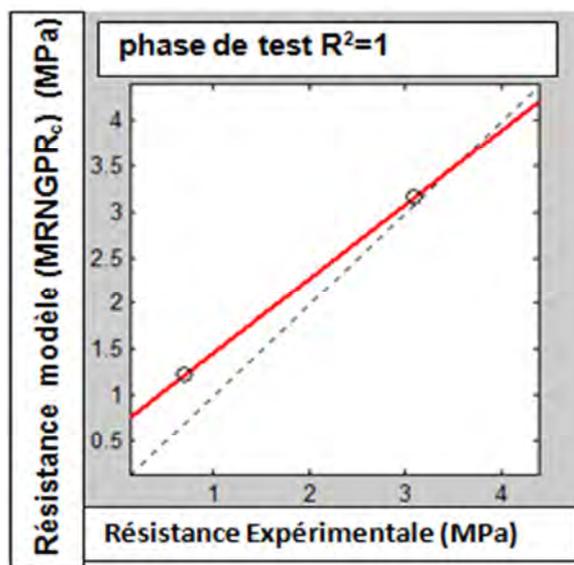


Figure (5-3) Performance du Modèle Global MRNGPR_c

5.5.3 Modèle Global du Résistance a la traction

Le tableau (5-4) récapitule l'ensemble des indications sur les paramètres requis et leurs plages de variations.

Tableau (5-4): Indicateurs utilisés – plage de variation.

Type de paramètres	Description	Plage de variation
Entrées	Clinker	19% < Clinker < 94%
	Laitier	0% < Laitier < 75%
	Gypse	Gypse=6%
	J : les jours	2 j , 28 j , 90 j
Sorties	Résistance à la traction Du béton	0.1 MPa < R _t < 4.4 MPa

Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt s'avère ici aussi le plus efficace et le plus performant.

L'analyse des résultats obtenus lors des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de taille RN 4-5-3-1 pour le modèle de prévision du

résistance à la traction du béton. L'architecture (topologie) du réseau à rétropropagation adopté est montrée sur la Figure (5-4).

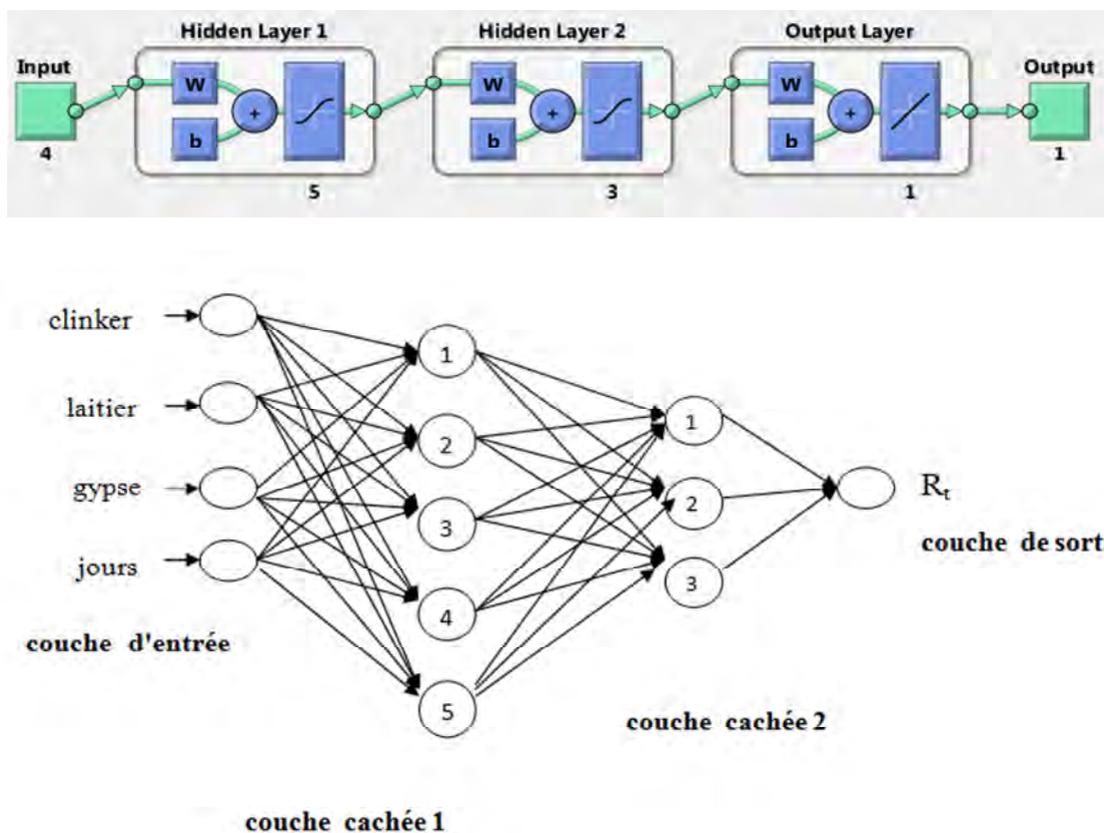


Figure (5-4) Architecture adoptée pour la résistance à la traction du béton en fonction du pourcentage de laitier .

L'apprentissage a été basé sur l'algorithme de Levenberg-Marquardt à Rétropropagation avec les fonctions de transfert standard Tansigmoïde et Purelinéaire correspondant respectivement aux couches cachées et de sortie, avec un taux d'apprentissage de 0.01 .

le tableau (5-5) et la figure(5-5),la figure (5-6) donnent les résultats des déférents phases d'application de modèle MRNGPR_t .

Tableau (5-5) : performance de modèle MRNGPR_t

Modèle RNA	Apprentissage	
	R2	MSE
MRNGPR _t (4-5-3-1)	0.999	1e-11

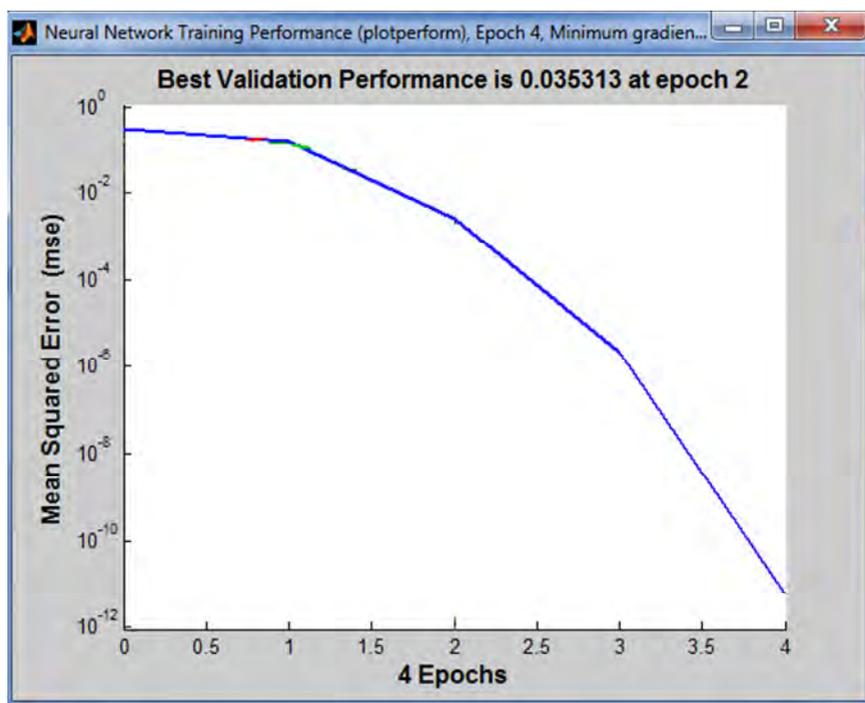
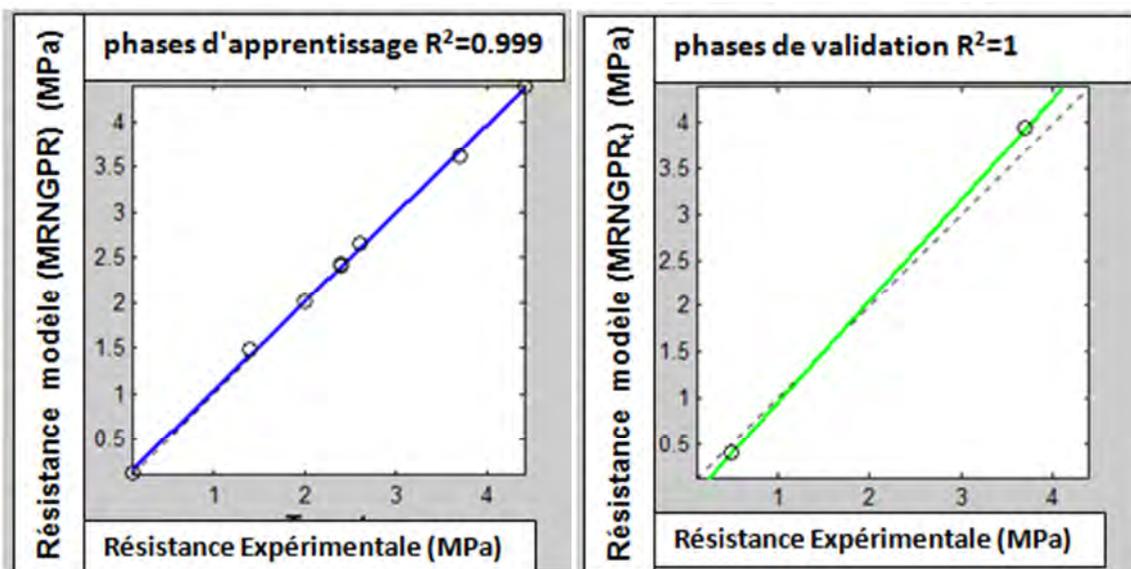


Figure (5-5) Apprentissage du modèle MRNGPR_t



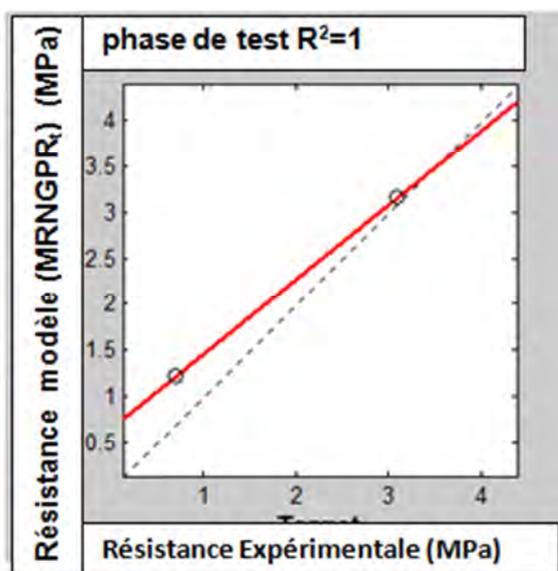


Figure (5-6) Performance du Modèle Global MRNGPR_t.

5.5.4 l'influence de la variation de la teneur en laitier

Le but de cette étude est de quantifier l'effet de la variation de la teneur en laitier en utilisant les modèles des réseaux de neurones (MRNGPR_c) et (MRNGPR_t).

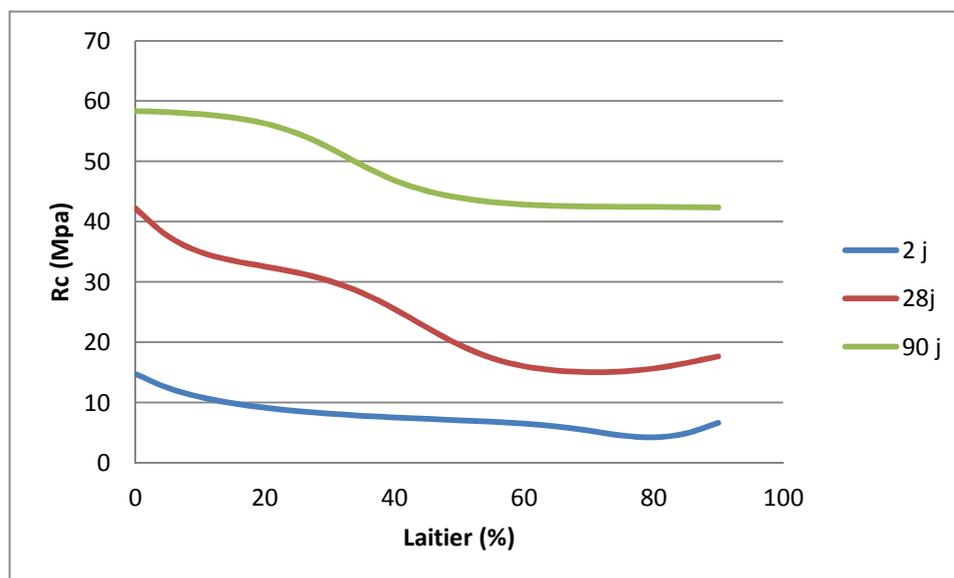


Figure (5-7) Influence de pourcentage de laitier sur les résistances à la compression.

d'après ces courbes on constate que pour tous les différents âge du béton, la résistance à la compression diminue d'une façon monotone avec l'augmentation du pourcentage de laitier, et que cette diminution est plus prononcée pour le béton d'âge 28j et 90j par rapport à 2 jours.

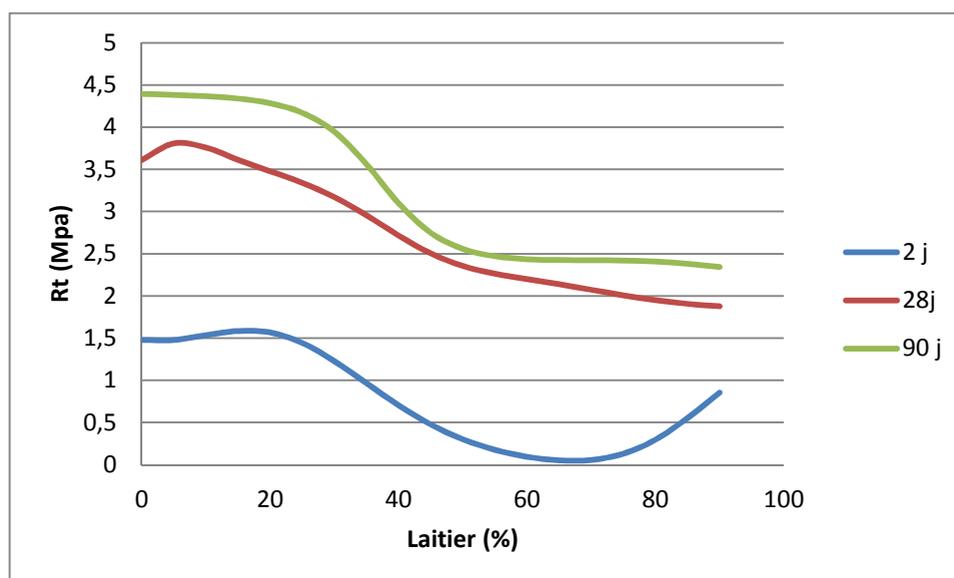


Figure (5-8) Influence de pourcentage de laitier sur les résistances à la traction.

on constate d'après la courbe de 2 jours que la résistance à la traction augmente légèrement avec l'augmentation du pourcentage du laitier jusqu'à la valeur de 20 %, au-delà de cette valeur la résistance à la traction diminue.

pour la courbe de 28 jours la résistance à la traction augmente avec l'augmentation du pourcentage du laitier jusqu'à la valeur de 5%, au-delà la résistance à la traction diminue d'une façon monotone.

en ce qui concerne la courbe de 90 jours, on constate que sa variation peut être divisée en 3 phases:

- ❖ phase 1 (0-25%): phase où la variation de la résistance à la traction a la tendance d'être stable avec l'augmentation du pourcentage du laitier.
- ❖ phase 2 (25-50%): phase où la résistance à la traction diminue d'une manière rapide avec l'augmentation du pourcentage du laitier
- ❖ phase 3 (50%): phase où la résistance à la traction diminue légèrement avec l'augmentation du pourcentage du laitier.

pour tous, la résistance mécanique en compression et en traction augmente avec le temps.

5.6 Conclusion

Cette étude a permis d'aboutir à un modèle décrivant l'évolution des variations du pourcentage de laitier dans le ciment au résistance mécanique à la compression et à la traction du béton.

CONCLUSION GENERALE

La composition chimique du clinker qui a servi de base pour la fabrication du ciment respecte dans une large mesure la fourchette admise en oxydes suivant: $\text{CaO-Fe}_2\text{O}_3\text{-SiO}_2\text{-Al}_2\text{O}_3\text{-MgO-K}_2\text{O-Na}_2\text{O-SO}_3$

Les résistances mécaniques des ciments sont déterminées par les essais sur mortier dit "normal", à 28 jours d'âges en traction et en compression des éprouvettes 4x4x16 cm. la résistance du mortier est alors considérée comme significative de la résistance du ciment.

Elle dépend de la composition chimique et la composition minéralogique.

La composition minéralogique plus signifie que la composition chimique du clinker, d'ailleurs le clinker est riche en C_2S et C_3S , ce qui influe directement sur l'augmentation de la résistance du ciment à court et à long terme.

la teneur en C_3S dans le ciment et caractérisé par une résistance mécanique élevée à jeune âge, et devient progressive avec une faible vitesse au cours du temps, par contre la teneur en C_2S caractérisé par une résistance mécanique faible à jeune âge, mais au cours du temps joue un rôle importante sur le développement de la résistance avec une vitesse de plus en plus forte que celle de C_3S .

Les alcalins sont très importants par leur influence sur les propriétés physique et mécanique du ciment .

Cette étude a permis d'aboutir à un modèle décrivant l'évolution des variations dimensionnelles des composition chimique et minéralogique de clinker au résistance mécanique, l'expansion et la consistance. Les modèles MRNGPR et MRNGPE et MRNGPC présentent l'avantage de prendre en considération tous les paramètres influant sur le résistance mécanique, l'expansion et la consistance.

La technique des réseaux de neurones présente l'avantage de pouvoir traiter un nombre illimité de paramètres caractéristiques du phénomène à la différence des méthodes statistiques. elle est d'un usage croissant en Génie Civil .

REFERENCES BIBLIOGRAPHIQUE

- [1] Sorin, F., L. Broussard, P. Roblin (2001). Régulation d'un processus industriel par réseaux de neurones, *Techniques de l'Ingénieur, traité Informatique industrielle*.
- [2] Jayet, A. (2002). Affective Computing: Apport des Processus Emotionnels aux Systèmes Artificiels.
Site, www.grappa.univ-lille3.fr/~torre/Recherche/Encadrement/Jayet2003/#sdfootnote10anc.
- [3] Dreyfus, G., J-M. Martinez, M. Samuelides, M.B. Gordon, F. Badran, and S.
- [4] Demartines, P (1994). Analyse de données par réseaux de neurones auto-organisés, *thèse de l'Institut National Polytechnique de Grenoble*.
- [5] Windrow, B., and M. E. Hoff (1960). Adaptive switching circuits. IRE Wescon Convention Record, part 4. 96-104.
- [6] Rosenblatt, F. (1958). The Perceptron : A probabilistic model for information storage and organization in the brain, *Phys. Rev.* **65**,386.
- [7] Golub, G.H., and D.P. O'Leary (1976). Some history of the conjugate gradient and Lanczos algorithms, *SIAM Review*, **31**,50-100
- [8] Battiti, R. (1992). First and Second Order Methods for Learning: Between Steepest Descent Methods and Newton's Method, *Neural Computation*, **4**, 141-166.
- [9] Oussar, Y. (1998). Réseaux d'ondelettes et réseaux de neurones pour la modélisation statique et dynamique de processus. *Thèse de doctorat de l'université pierre et marie curie*.
- [10] thiria S., Lechevallier y., Gascuel O. et Canu S., "statique et méthode neuronales ", Edition Dunod, Paris, France, 1997
- [11] flood I, "next generation artificial neural network for civil Engineering ", journal of computing in Civil Engineering, September - October 2006, pp.305-307.
- [12] W. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics* 5: 115-133, 1943.
- [13] D. Hebb, *The Organization of Behavior*, New York: Wiley, 1949.
- [14] M. Minsky and S. Papert, *Perceptrons*, Expanded Edition. MIT Press, 1988.
- [15] D. Ackley, G. Hinton and T. Sejnowski, "A learning algorithm for Boltzmann machines," *Cognitive Science* 9 : 147-169, 1985.
- [16] D. Rumelhart, G. Hinton & R. Williams, "Learning internal representations by error propagation," *Parallel Distributed Processing, Vol. 1.*, D. Rumelhart and J. McClelland Eds. Cambridge: MIT Press, pp. 318-362, 1986.

- [17] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities," Proceedings of the National Academy of Sciences 79 : 2554-2558, 1982.
- [18] W. James, *Psychology (Briefer Course)*, New York: Holt, Chapter XVI, "Association", pp. 253-279, 1890.
- [19] Najjar Y., Basheer I.A., Hajmeer M.N., « Computational neural networks for predictive microbiology : Methodology », International Journal of Food Microbiology, N° 34, 1997, pp. 27-49
- [20] Fortin V., Ouarda T., Rasmussen T.P., et Bobée B., « Revue bibliographique des méthodes de prévision des débits », Revue des Sciences de l'Eau, N° 4, 1997, pp. 461-487.
- [21] McCulloch W.S., Pitts W., « A logical Calculus of the Ideas Imminent in Nervous Activity », Bulletin of Mathematical Biophysics, N° 5, 1943, pp. 115-133.
- [22] Senthil Kumar A.R., Sudheer K.P., Jain S.K., and Agarwal P.K., « Rainfall-runoff modelling using artificial neural networks : Comparison of network types », Hydrological Processes, 2004
- [23] Mas J.F., Puig H., Palacio J.L., and Sosa Lopez A., « Modeling deforestation using GIS and artificial neural networks », Environmental Modeling and Software, N° 19, 2004, pp. 461-471.
- [24] Haykin S., « Neural Networks », A comprehensive foundation, Macmillan College, Publishing Company, New York, 1994.
- [25] Lippmann R.P., « An Introduction to Computing with Neural Nets », IEEE ASSP Magazine, April 1987, Vol. 4, N° 2, pp. 4-22.
- [26] Najjar Y., Zhang X., « Characterizing the 3D Stress-Strain Behavior of sandy Soils: A Neuro- Mechanistic Approach », In Numerical Methods in Geotechnical Engineering, ASCE Geotechnical Special Publication, N° 96, Filz G. and Griffiths D. (Editors), 2000, pp. 43-57.
- [27] Basheer I.A., Hajmeer M.N., « Artificial neural network : Fundamentals, Computing Design, and Application », Journal of Microbiological Methods, N° 34, 1997, pp. 3-31.
- [28] Senthil Kumar A.R., Sudheer K.P., Jain S.K., and Agarwal P.K., « Rainfall-runoff modelling using artificial neural networks : Comparison of network types », Hydrological Processes, 2004.
- [29] Hecht-Nielsen R., « Solution for a distributed hydrological model and applications », Neurocomputing, Addison-Wesley, Reading, Massachusetts, USA, 1990, pp. 89-93.
- [30] Ali Bouasida Loucif " effet de la qualité du ciment sur les propriété mécanique du béton ". thèse de magistère , institut de génie civil center universitaire de Tebessa, 2002.

[31] Gutteche M.N " contribution à la valorisation des laitiers de haut-fourneau d'elhadjar dans le domaine de construction caractérisation des matériaux élaboré " . thèse de doctorat d'état en génie civil , faculté des science de l'ingénieure département de génie civil université Mentouri Constantine,2002 .

1. Clinker (Composition chimiques et minéralogiques)

N°	SiO ₂	Al ₂ O ₃	Fe ₂ O ₃	CaO	MgO	Na ₂ O	K ₂ O	LSF	MS	MA	Rc(mortier)			C ₃ S	C ₂ S	C ₃ A	C ₄ AF	PAF	T	Con	Exp
											2j	7j	28j								
1	22,4	5,47	3,35	64,6	1,65	0,17	0,83	90,63	2,54	1,63	22	36	48	51,83	25,68	8,9	10,28	0,9	60	27	2,5
2	22,2	5,49	3,32	64,1	1,59	0,12	0,81	90,69	2,52	1,65	23	37	50	51,24	25,38	9	10,17	0,7	60	28	1,8
3	21,9	5,44	3,31	64,3	1,64	0,13	0,77	91,86	2,51	1,64	22	35	48	54,02	22,56	8,87	10,12	0,62	60	27	2,5
4	22	5,33	3,21	64,5	1,62	0,41	0,78	92,36	2,57	1,66	21	36	49	55,66	21,39	8,76	9,82	0,65	60	28	1,8
5	21,5	5,4	3,23	64,6	1,62	0,13	0,77	94,04	2,49	1,67	24	37	50	58,9	17,61	8,89	9,89	0,54	60	27	2
6	21,5	5,4	3,23	64,6	1,62	0,13	0,77	94,04	2,49	1,67	19	33	46	58,9	17,61	8,89	9,89	0,54	60	27	2,5
7	21,5	5,4	3,23	64,6	1,62	0,13	0,77	94,04	2,49	1,67	18	33	46	58,9	17,61	8,89	9,89	0,54	60	27	2,5
8	21,9	5,48	3,13	65,5	1,57	0,12	0,79	94,03	2,54	1,75	21	37	48	59,67	18,03	9,27	9,57	0,56	60	28	1,6
9	22	5,51	3,29	64,9	1,65	0,19	0,75	92,5	2,51	1,7	25	40	50	55,56	21,31	9,14	9,89	0,3	60	26	2,8
10	22	5,35	3,19	64,7	1,64	0,13	0,76	93,67	2,54	1,68	24	38	50	58,4	18,42	8,82	9,73	0,42	60	27	3
11	21,7	5,33	3,26	64,5	1,68	0,27	0,81	92,93	2,54	1,63	26	41	49	56,81	20,02	8,67	9,98	0,6	65	26	1,5
12	21,8	5,7	2,93	65,6	1,74	0,2	0,8	94,01	2,53	1,95	25	38	50	58,82	18,59	10,21	8,95	0,56	60	28	1,3
13	21,8	5,7	2,93	65,6	1,74	0,2	0,8	94,01	2,53	1,95	24	36	48	58,82	18,59	10,21	8,95	0,56	60	29	2,3
14	21,7	5,46	3,18	65,5	1,68	0,19	0,78	94,55	2,51	1,78	24	36	48	59,12	17,58	8,92	9,67	1	60	26	2
15	22,1	5,74	3,31	65,9	1,66	0,19	0,84	92,97	2,44	1,74	24	36	47	57,88	20,68	9,73	10,15	1,02	60	26	2,6
16	22	5,49	3,22	65,5	1,64	0,19	0,75	93,6	2,54	1,68	21	35	46	59,05	18,83	8,94	9,87	0,66	65	27	1,8
17	21,5	5,57	2,99	64,3	1,75	0,23	0,88	93,82	2,51	1,86	21	38	49	54,42	18,5	9,75	9,13	0,47	60	26	1,6
18	22,5	5,63	3,36	66,6	1,77	0,2	0,79	92,93	2,5	1,67	21	37	48	58,42	20,9	9,31	10,31	0,9	60	26	2,5
19	22,5	5,63	3,36	66,6	1,77	0,2	0,79	92,93	2,5	1,67	23	39	50	58,42	20,9	9,31	10,31	0,9	65	26	2,3
20	22,5	5,63	3,36	66,6	1,77	0,2	0,79	92,93	2,5	1,67	23	39	50	58,42	20,9	9,31	10,31	0,9	65	26	2,3
21	22,2	5,57	3,35	66	1,65	0,14	0,77	93,14	2,47	1,66	22	37	48	58,75	20,32	9,25	10,34	1,51	60	26	1,6
22	22,2	5,77	3,32	66	1,69	0,2	0,86	92,75	2,44	1,74	23	39	51	56,99	21,49	9,78	10,22	1,16	65	27	1,6
23	21,8	5,36	3,28	65,5	1,65	0,22	0,78	94,24	2,52	1,64	21	38	50	58,88	18,52	8,99	10,01	1,26	60	27	1,8
24	21,9	5,48	3,32	65,4	1,68	0,18	0,8	93,5	2,48	1,65	21	38	51	58,9	18,42	9,12	9,82	0,47	60	27	1,3
25	22,4	5,59	3,19	64,9	1,67	0,2	0,89	90,84	2,55	1,75	24	40	51	51,94	25,5	9,48	9,75	0,6	65	26	2,5

2. Clinker (Composition chimiques et minéralogiques)

N°	SiO ₂	Al ₂ O ₃	Fe ₂ O ₃	CaO	MgO	Na ₂ O	K ₂ O	LSF	MS	MA	Rc(mortier)			C ₃ S	C ₂ S	C ₃ A	C ₄ AF	PAF	T	Con	Exp
											2j	7j	28j								
26	22,4	5,59	3,19	64,9	1,67	0,2	0,89	90,84	2,55	1,75	22	36	49	51,94	25,5	9,48	9,75	0,6	60	28	2
27	22,4	5,59	3,19	64,9	1,67	0,2	0,89	90,84	2,55	1,75	22	36	49	51,94	25,5	9,48	9,75	0,6	60	28	2
28	22,5	5,74	3,33	64,9	1,77	0,2	0,81	90,05	2,48	1,72	24	40	49	50,03	27,96	9,7	10,25	1,23	60	26	1,5
29	22	5,52	3,45	64,1	1,67	0,21	0,85	91,2	2,45	1,6	24	40	50,3	51,92	23,95	8,81	10,51	0,21	60	27	1,5
30	22,1	5,59	3,38	65,1	1,73	0,21	0,84	91,87	2,47	1,66	21	38	51	54,28	22,88	9,16	10,31	0,47	65	27	2,8
31	21,8	5,4	3,19	65,2	1,68	0,21	0,79	94,01	2,53	1,69	20	38	51	59,57	17,85	9,97	9,75	0,5	60	28	2,3
32	20,7	5,06	2,95	61,6	1,48	0,18	0,76	93,5	2,59	1,72	20	36	49	56,1	18,21	8,58	9,12	1,84	60	28	3
33	20,7	5,06	2,95	61,6	1,48	0,18	0,76	93,5	2,59	1,72	18	34	48	56,1	18,21	8,58	9,12	1,84	55	28	2
34	21,7	5,4	3,18	65	1,68	0,2	0,78	93,98	2,53	1,7	19	34	47	59,23	17,86	8,99	9,71	0,53	60	27	2,1
35	21,8	5,66	3,28	65,5	1,66	0,19	0,79	93,64	2,44	1,7	20	35	48	56,46	20,44	9,25	10,47	0,54	65	28	1,8
36	22,6	5,6	3,52	64,6	1,61	0,22	0,85	89,55	2,48	1,59	19	33	47	49,01	28,38	8,99	10,79	0,91	55	27	2
37	21,9	5,49	3,25	64,4	1,64	0,18	0,75	92	2,51	1,69	19	37	49	54,11	22,37	9,11	9,92	0,5	60	27	1,3
38	22,2	5,57	3,19	65,5	1,64	0,19	0,79	92,53	2,53	1,74	18	33	47	56,34	21,62	9,42	9,78	0,54	60	27	2
39	22,2	5,57	3,19	65,5	1,64	0,19	0,79	92,53	2,53	1,74	20	33	46	56,34	21,61	9,42	9,78	0,54	60	28	2,8
40	22,1	5,7	3,29	65	1,69	0,19	0,76	91,92	2,46	1,73	22	36	47	54,24	23,09	9,63	10,11	0,99	65	27	1,6
41	21,9	5,65	3,27	65,2	1,65	0,19	0,8	93,11	2,45	1,73	23	38	49	56,94	20,24	9,52	10,03	0,77	60	26	1,8
42	22	5,68	3,2	65,8	1,64	0,18	0,81	93,48	2,48	1,78	22	36	48	58,21	19,61	9,72	9,79	0,68	65	27	1,2
43	22	5,79	3,27	65,7	1,61	0,18	0,64	93,09	2,43	1,77	24	37	48	57,04	20,62	9,9	10,02	0,8	60	25	1,8
44	22	5,79	3,13	65,6	1,61	0,19	0,78	93,13	2,47	1,85	23	36	46	56,81	20,57	10,1	9,56	0,52	60	26	2,6
45	22	5,79	3,13	65,6	1,61	0,19	0,78	93,13	2,47	1,85	23	36	47	56,81	20,57	10,1	9,56	0,52	60	26	2,5
46	21,9	5,48	3,05	65,5	1,87	0,19	0,81	93,84	2,57	1,8	24	37	47	59,08	18,48	9,4	9,31	0,32	55	26	1,5
47	22,4	5,49	2,98	65,6	1,63	0,19	0,8	92,16	2,64	1,84	23	39	49	55,71	22,54	9,56	9,11	0,5	60	26	3
48	21,6	5,41	3,14	63,8	1,6	0,21	0,8	92,59	2,52	1,72	22	37	48	54,66	20,67	9,02	9,54	0,44	60	26	1,6
49	21,5	5,26	3,15	64	1,64	0,21	0,75	93,5	2,55	1,67	19	37	48	57,23	18,44	9,17	9,61	0,36	60	25	2
50	21,4	5,23	3,23	64,4	1,59	0,21	0,77	94,44	2,52	1,62	19	37	48	59,7	16,3	9,4	9,82	0,6	60	26	1,8

3. Clinker (Composition chimiques et minéralogiques)

N°	SiO ₂	Al ₂ O ₃	Fe ₂ O ₃	CaO	MgO	Na ₂ O	K ₂ O	LSF	MS	MA	Rc(mortier)			C ₃ S	C ₂ S	C ₃ A	C ₄ AF	PAF	T	Con	Exp
											2j	7j	28j								
51	21,4	5,23	3,23	64,4	1,59	0,21	0,77	94,44	2,52	1,62	22	39	50	59,7	16,3	9,4	9,82	0,6	60	26	2
52	21,4	5,25	3,3	64,1	1,57	0,2	0,78	93,9	2,5	1,6	21	38	50	58,25	17,4	8,33	10,03	0,6	55	26	2,8
53	21,4	5,36	3,44	64,7	1,62	0,21	0,77	94,5	2,43	1,56	21	37	47	59,75	16,26	8,38	10,45	0,48	60	25	2,3
54	21,5	5,48	3,29	64,3	1,56	0,21	0,78	93,54	2,45	1,67	22	38	49	57,19	18,73	9,01	10,06	0,42	65	25	1,6
55	21,6	5,56	3,13	64	1,51	0,21	0,76	92,57	2,49	1,78	22	38	49	54,59	21,15	9,48	9,56	0,42	60	26	1,6
56	28	5,6	3,22	63,9	1,74	0,21	0,77	91,55	2,48	1,74	22	37	49	52,36	23,51	9,45	9,85	0,59	60	25	1,6
57	28	5,6	3,22	63,9	1,74	0,21	0,77	91,55	2,48	1,74	21	37	47	52,36	23,51	9,45	9,85	0,59	65	25	2,8
58	28	5,6	3,22	63,9	1,74	0,21	0,77	91,55	2,48	1,74	21	36	45	52,36	23,51	9,45	9,85	0,59	65	25	1,8
59	21,5	5,5	3,16	63,4	1,6	0,2	0,77	92,39	2,48	1,74	21	37	47	53,73	21,27	9,27	9,66	0,4	60	25	1
60	21,2	5,26	3,17	63,9	1,6	0,21	0,77	94,6	2,51	1,66	18	33	48	59,6	16,08	8,64	9,69	0,6	65	30	2,3
61	22	5,6	3,2	64,3	1,57	0,21	0,78	91,43	2,5	1,75	17	32	46	52,52	23,92	9,49	9,81	0,77	55	27	1,5
62	21,4	5,57	3,29	64	1,57	0,1	0,76	93,26	2,41	1,69	20	36	48	55,97	19,42	9,23	10,06	0,5	60	26	1,5
63	21,7	5,57	3,12	64,4	1,56	0,1	0,78	92,7	2,5	1,79	21	38	49	55,25	20,96	9,53	9,52	0,5	60	25	1
64	21,8	5,68	3,21	65,1	1,55	0,15	0,87	93,33	2,45	1,77	20	37	48	57,17	19,8	9,7	9,83	0,78	60	25	1,6
65	21,3	5,46	3,31	63,9	1,58	0,47	0,82	93,42	2,43	1,65	23	41	50	56,47	18,84	8,9	10,1	0,33	60	28	1,6
66	21,3	5,46	3,1	63,1	1,55	0,13	0,75	92,55	2,49	1,76	21	40	51	53,77	20,84	9,27	9,47	0,42	55	28	1,1
67	21,9	5,54	3,12	64,4	1,69	0,13	0,78	92,06	2,51	1,74	20	36	48	53,95	22,32	9,34	9,71	0,37	65	27	2,1
68	21,9	5,54	3,12	64,4	1,69	0,13	0,78	92,06	2,51	1,74	21,75	36,1	47,25	53,95	22,32	9,34	9,71	0,37	55	26	3,5
69	21,7	5,42	3,24	64,9	1,63	0,12	0,75	92,66	2,51	1,67	23	38	48	58,29	18,51	8,91	9,87	0,3	60	25	2
70	21,7	5,36	3,22	64,5	1,62	0,11	0,76	93,3	2,52	1,66	23	38	48	57,29	19,2	8,79	9,84	0,4	60	25	2
71	21,6	5,49	3,26	64,3	1,63	0,11	0,77	93,03	2,47	1,62	24	40	50	56,17	19,95	9	9,96	0,5	60	25	2,8
72	21,7	5,55	3,24	64,8	1,6	0,14	0,77	93,41	2,49	1,72	23	37	48	57,51	19,29	9,28	9,86	0,7	60	25	1,1
73	21,3	5,48	3,06	65,1	1,64	0,16	0,75	95,58	2,5	1,79	23	36	47	62,11	14,53	9,39	9,34	0,45	55	25	3,1
74	21,3	5,48	3,06	65,1	1,64	0,16	0,75	95,58	2,5	1,79	23	37	47	62,11	14,53	9,39	9,34	0,45	55	25	3,8
75	21,2	5,34	3,16	65,6	1,65	0,14	0,8	96,7	2,5	1,69	20	38	48	65,49	11,87	8,86	9,69	0,56	60	26	1,8

4. Clinker (Composition chimiques et minéralogiques)

N°	SiO ₂	Al ₂ O ₃	Fe ₂ O ₃	CaO	MgO	Na ₂ O	K ₂ O	LSF	MS	MA	Rc(mortier)			C ₃ S	C ₂ S	C ₃ A	C ₄ AF	PAF	T	Con	Exp
											2j	7j	28j								
76	20,9	5,48	3,29	63,9	1,62	0,14	0,73	95,31	2,38	1,67	21	37	48	60,09	14,69	8,9	10,02	0,24	60	26	1
77	21,3	5,39	3,02	64,2	1,63	0,14	0,77	94,54	2,53	1,78	21	36	46	59,26	16,57	9,21	9,22	0,35	60	26	2,3
78	21,5	5,55	3,11	64,1	1,64	0,14	0,76	93,22	2,48	1,79	20	36	47	55,97	19,66	9,5	9,47	0,35	60	25	2
79	21,7	5,6	3,16	64,1	1,61	0,14	0,79	92,25	2,48	1,78	22	37	49	53,82	21,92	9,56	9,64	0,44	60	25	2,6
80	21,7	5,6	3,16	64,1	1,61	0,14	0,79	92,25	2,48	1,78	23,2	37,2	47,5	53,82	21,92	9,56	9,64	0,44	60	25	2
81	21,6	5,57	3,07	63,6	1,65	0,15	0,75	92,54	2,49	1,82	21,2	36	45,9	54,16	21,2	9,62	9,39	0,35	60	26,2	1,5
82	21,2	5,47	3,06	64,2	1,62	0,14	0,79	94,8	2,48	1,79	21,6	36,8	47,9	59,54	16,1	9,38	9,37	0,57	60	24,7	1,5
83	20,9	5,51	3,09	64	1,62	0,17	0,8	95,52	2,43	1,79	21,1	37,8	48,6	60,53	14,51	9,43	9,42	0,4	55	25,4	1,5
84	22	5,54	3,11	63,6	1,67	0,17	0,85	90,47	2,55	1,78	17,9	33,6	45,2	49,85	25,93	9,46	9,51	0,5	55	26,1	1,5
85	21,4	5,65	3,04	64,4	1,7	0,14	0,77	93,73	2,47	1,86	21,8	37,8	47,6	57,02	18,78	9,88	9,3	0,5	55	25,2	1
86	21,4	5,65	3,04	64,4	1,7	0,14	0,77	93,73	2,47	1,86	23	37	47,4	57,02	18,78	9,88	9,3	0,5	60	24,3	1,5
87	21,6	5,63	3,02	64,6	1,69	0,15	0,79	93,42	2,5	1,87	22,4	37,2	47,3	56,79	19,55	9,88	9,23	0,6	55	24,9	1,5
88	21,1	5,48	2,97	64,9	1,71	0,15	0,73	96,07	2,5	1,85	24,1	37,4	46,7	62,82	13,51	9,56	9,07	0,47	60	24,9	1,5
89	20,4	5,35	3,03	64,2	1,6	0,14	0,74	96,54	2,44	1,77	24,5	38,2	47,7	62,12	12,1	9,12	9,25	0,57	60	25	0,8
90	21	5,33	3,09	63,8	1,62	0,14	0,77	94,78	2,48	1,74	22	37,1	47,1	59,42	15,93	9,1	9,47	0,7	60	25,9	1
91	21,3	5,42	3,37	64,5	1,68	0,17	0,73	94,4	2,42	1,61	23,8	38	48,2	59,34	16,65	8,71	10,3	0,46	60	24,8	1,25
92	21,3	5,42	3,37	64,5	1,68	0,17	0,73	94,4	2,42	1,61	21	35,6	47,6	59,34	16,65	8,71	10,3	0,46	65	26,2	1,75
93	21,3	5,54	3,45	64,5	1,73	0,14	0,8	94,26	2,37	1,6	19,1	36,2	46,6	58,81	17,04	8,89	10,56	0,5	60	25,6	0,9
94	21,1	5,44	3,17	64,6	1,55	0,14	0,74	95,45	2,47	1,75	20,8	37,1	48,3	61,47	14,45	9,19	9,5	0,42	60	26,1	2,25
95	21,3	5,63	3,29	64	1,74	0,14	0,75	93,71	2,38	1,71	19,8	34,4	44,8	56,61	18,48	9,39	10,04	0,35	55	25,7	1,6
96	20,3	5,27	2,86	65,6	1,7	0,14	0,8	95,25	2,61	1,91	23,2	38,7	47,5	57,07	18,8	9,65	8,69	0,4	60	25,2	1
97	20,6	5,16	3,11	63,7	1,77	0,14	0,66	96,93	2,49	1,66	23,7	37,6	49	64	10,95	8,45	9,49	0,37	60	25,5	1,3
98	21	5,14	2,7	64	1,54	0,12	0,75	95,98	2,68	1,91	23,6	35,2	47,7	62,53	13,35	9,11	8,23	0,37	60	25,6	1,5
99	20,7	5,38	3,08	63,2	1,5	0,17	0,71	95,16	2,45	1,75	23,6	37,1	48,3	59,39	15	9,11	9,42	0,46	60	25,3	3
100	20,7	5,38	3,08	63,2	1,5	0,17	0,71	95,16	2,45	1,75	22	37,3	48,9	59,39	15	9,11	9,42	0,5	60	25,2	2

5. Clinker (Composition chimiques et minéralogiques)

N°	SiO ₂	Al ₂ O ₃	Fe ₂ O ₃	CaO	MgO	Na ₂ O	K ₂ O	LSF	MS	MA	Rc(mortier)			C ₃ S	C ₂ S	C ₃ A	C ₄ AF	PAF	T	Con	Exp
											2j	7j	28j								
101	20,9	5,54	3,4	63,3	1,59	0,14	0,74	94,19	2,33	1,63	24,9	39,7	51,3	57,06	16,8	8,96	10,39	0,35	55	25,1	0,8
102	21,1	5,35	3,15	63,8	1,65	0,14	0,71	94,5	2,49	1,7	23	38	51	58,92	16,43	8,91	9,61	0,47	60	25	1,3
103	21,2	5,39	3,21	64	1,62	0,12	0,69	94,24	2,47	1,68	23	40	52	58,53	17,05	8,91	9,8	0,46	60	26	1,3
104	20,9	5,19	3,13	63	1,64	0,09	0,72	94,26	2,52	1,66	23	39	49	57,99	16,57	8,49	9,55	0,43	40	24	0,7
105	21,2	5,32	3,02	63,2	1,58	0,11	0,74	93,4	2,55	1,76	24	40	52	56,11	18,8	9,03	9,21	0,43	55	25	1,6
106	21,2	5,37	3,06	63,4	1,6	0,08	0,78	93,58	2,52	1,75	22	38	48	56,5	18,498	9,09	9,33	0,4	60	25	1,5
107	21,1	5,29	3,13	62,9	1,6	0,09	0,78	93,38	2,51	1,69	22	39	49	55,99	18,67	8,77	9,57	0,6	55	25	1,2
108	21,5	5,56	3,23	63,5	1,7	0,1	0,63	92,13	2,45	1,72	24	40	49	53,17	21,94	9,31	9,88	0,5	60	25	0,8
109	21,2	5,71	2,92	63,9	1,65	0,18	0,73	94,93	2,62	1,77	24	36	46	60,47	15,49	8,8	8,91	0,51	60	25	1,5
110	21,2	5,71	2,92	63,9	1,65	0,18	0,73	94,93	2,62	1,77	24	40	48	60,47	15,49	8,8	8,91	0,51	55	25	2,3
111	21,3	5,8	2,94	63,9	1,66	0,18	0,73	95,03	2,6	1,78	26	39	47	61,2	15,54	8,48	8,94	0,72	60	25	1,3
112	20,9	5,31	3,27	64,3	1,67	0,63	0,71	96,15	2,43	1,62	25	39	48	62,89	12,76	8,61	10,01	0,57	60	24	2
113	21,6	5,36	3,2	63,2	1,62	0,32	0,74	91,7	2,52	1,68	25	40	49	52,91	22,7	8,89	9,83	1,02	60	24	1,8
114	21,5	5,07	3,2	63,3	1,58	0,26	0,78	92,83	2,6	1,58	23	40	49	56,24	19,66	8,09	9,81	0,85	60	25	1
115	21,7	5,23	3,24	63,7	1,6	0,26	0,73	92,21	2,56	1,61	21	37	47	55,19	21,51	8,52	10	1,46	60	25	1
116	21,7	5,23	3,24	63,7	1,6	0,26	0,73	92,21	2,56	1,61	23	38	47	55,19	21,51	8,52	10	1,46	60	25	0,8
117	21,6	5,13	3,2	63	1,58	0,24	0,72	91,91	2,59	1,6	21	37	46	54,26	21,94	8,31	9,89	1,56	60	24	1
118	20,6	5,06	3,3	63,1	1,56	0,16	0,73	96,03	2,46	1,53	23	38	47	62,73	12,58	7,95	10,2	1,59	60	24	1,3
119	20,8	5,22	3,32	63,8	1,61	0,25	0,72	95,82	2,44	1,57	24	39	48	61,89	13,18	8,24	10,13	0,27	60	25	1,1
120	21,2	5,34	3,24	63,7	1,57	0,16	0,74	94,11	2,47	1,65	23	38	48	58,71	17,34	8,81	9,98	1,41	55	24	1
121	21,7	5,37	3,1	63,7	1,55	0,21	0,75	92,19	2,56	1,73	23	38	47	54,47	21,84	9,11	9,55	1,2	60	24	1
122	21,7	5,37	3,1	63,7	1,55	0,21	0,75	92,19	2,56	1,73	21	34	44	54,47	21,84	9,11	9,55	1,2	60	24	2,3
123	21,7	5,37	3,1	63,7	1,55	0,21	0,75	92,19	2,56	1,73	21	35	46	54,47	21,84	9,11	9,55	1,2	60	25	2,7
124	21,7	5,36	3,54	64,9	1,6	0,17	0,75	93,4	2,44	1,51	22	37	48	58,49	18,87	8,3	10,89	1,02	60	24	1
125	21,7	5,41	3,23	64,3	1,59	0,22	0,76	92,95	2,51	1,67	22	35	48	56,52	20,13	8,96	9,92	1,02	60	24	1,6

6. Clinker (Composition chimiques et minéralogiques)

N°	SiO ₂	Al ₂ O ₃	Fe ₂ O ₃	CaO	MgO	Na ₂ O	K ₂ O	LSF	MS	MA	Rc(mortier)			C ₃ S	C ₂ S	C ₃ A	C ₄ AF	PAF	T	Con	Exp
											2j	7j	28j								
126	21,3	5,16	3,32	65	1,66	0,36	0,82	95,76	2,51	1,55	22	38	49	63,92	13,5	8,15	10,21	1,02	55	25	0,6
127	21,7	5,46	3,18	65,5	1,68	0,19	0,78	94,55	2,51	1,78	20	33	44	59,12	17,58	8,92	9,67	0,48	60	25	1,8
128	21,8	5,45	3,26	64,9	1,7	0,24	0,77	93,39	2,5	1,67	21	36	47	57,83	19,22	9	9,97	0,6	65	25	0,8
129	21,6	5,63	3,02	64,6	1,69	0,15	0,73	93,43	2,5	1,87	23,3	38,7	47,1	56,79	19,55	9,88	9,23	0,83	65	25	1
130	21,5	5,39	3,32	63,7	1,67	0,36	0,83	92,85	2,46	1,62	24,2	36	47,7	55,6	20,04	9,73	10,17	0,61	60	25	0,3
131	21	5,42	3,11	61,7	1,55	0,5	0,81	91,97	2,45	1,74	25,7	32,7	48,9	51,37	21,17	9,57	9,52	0,68	55	24	1
132	21	5,42	3,11	61,7	1,55	0,5	0,81	91,97	2,45	1,74	23	33	43,5	51,37	21,17	9,57	9,52	0,68	60	25	2
133	20,9	5,29	3,09	61,1	1,44	0,24	0,74	91,74	2,49	1,71	20,6	35,5	42,5	50,68	21,93	8,84	9,44	0,59	55	25	2
134	21,1	5,43	3,05	63	1,59	0,22	0,75	93,28	2,49	1,76	20,5	36	43,4	55,22	19,16	9,28	9,31	0,42	50	25	1,5
135	21,2	5,43	3,12	62,2	1,52	0,27	0,77	91,61	2,43	1,74	22,3	35,6	44,5	51,07	22,7	9,18	9,52	0,55	60	25	2
136	21,7	5,37	3,1	63,7	1,55	0,21	0,76	92,19	2,56	1,73	22,2	36,1	46	54,45	21,84	9,11	9,55	0,42	60	25	1,5
137	21,7	5,37	3,1	63,7	1,55	0,21	0,76	92,19	2,56	1,73	24	35,3	47,3	54,45	21,84	9,11	9,55	0,42	60	25	1,8
138	20,72	5,46	3,89	63	1,57	0,15	0,71	94,05	2,22	1,42	21,3	35	46,3	54,9	17,61	7,89	11,82	1,02	55	25	1,5
139	21,6	5,33	3,28	62,6	1,6	0,15	0,71	90,78	2,51	1,62	21,8	35	45,8	50,26	24,44	8,62	10,03	0,52	60	25	2
140	20,8	5,28	3,32	62,1	1,57	0,16	0,71	93,09	2,42	1,59	19,4	34,1	46	54,44	18,91	8,42	10,15	0,39	60	25	1,5
141	21,92	5,55	3,37	64,07	1,57	0,16	0,71	91,34	2,46	1,65	19,9	35	43,7	49,5	25,1	9,01	10,24	1,26	60	25	1

Les résistances à la compression et à la traction du béton en fonction de la variation des pourcentage de clinker et laitier

N	Clinker %	Laitier %	Gypse %	Jours	Résistance (MPa)	
					compression	traction
1	94	0	6	2	14,76	1,4771
2	89	5	6	2	12,4233	1,4775
3	84	10	6	2	10,9097	1,534
4	79	15	6	2	9,8774	1,5845
5	74	20	6	2	9,1327	1,5668
6	69	25	6	2	8,5703	1,4431
7	64	30	6	2	8,1321	1,2272
8	59	35	6	2	7,7833	0,96642
9	54	40	6	2	7,5	0,70447
10	49	45	6	2	7,2607	0,47641
11	44	50	6	2	7,0409	0,30094
12	39	55	6	2	6,8044	0,1762
13	34	60	6	2	6,4928	0,093738
14	29	65	6	2	6,0217	0,051008
15	24	70	6	2	5,3254	0,056415
16	19	75	6	2	4,5384	0,12998
17	14	80	6	2	4,1942	0,29505
18	9	85	6	2	4,8617	0,5518
19	4	90	6	2	6,5953	0,85379
20	94	0	6	28	42,23	3,6095
21	89	5	6	28	37,6366	3,8071
22	84	10	6	28	34,9728	3,7577
23	79	15	6	28	33,5393	3,613
24	74	20	6	28	32,5521	3,4775
25	69	25	6	28	31,5331	3,34
26	64	30	6	28	30,1574	3,1707
27	59	35	6	28	28,175	2,9559
28	54	40	6	28	25,5097	2,7168

29	49	45	6	28	22,4431	2,5045
30	44	50	6	28	19,5635	2,3552
31	39	55	6	28	17,3707	2,2629
32	34	60	6	28	15,9975	2,1987
33	29	65	6	28	15,2934	2,1383
34	24	70	6	28	15,0379	2,0727
35	19	75	6	28	15,1319	2,0071
36	14	80	6	28	15,6245	1,9501
37	9	85	6	28	16,5267	1,9065
38	4	90	6	28	17,6253	1,8761
39	94	0	6	90	58,3393	4,3926
40	89	5	6	90	58,1495	4,3824
41	84	10	6	90	57,8242	4,3667
42	79	15	6	90	57,2605	4,3382
43	74	20	6	90	56,2847	4,2826
44	69	25	6	90	54,6578	4,1702
45	64	30	6	90	52,2422	3,9461
46	59	35	6	90	49,3782	3,5617
47	54	40	6	90	46,8476	3,1001
48	49	45	6	90	45,0877	2,7489
49	44	50	6	90	43,9614	2,5577
50	39	55	6	90	43,2441	2,4698
51	34	60	6	90	42,8203	2,4349
52	29	65	6	90	42,6037	2,4249
53	24	70	6	90	42,5048	2,4234
54	19	75	6	90	42,4554	2,4194
55	14	80	6	90	42,421	2,4059
56	9	85	6	90	42,3899	2,3799
57	4	90	6	90	42,3601	2,3436