

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE



Département d'hydraulique

Mémoire de Master

En Vue de l'Obtention du Diplôme de Master en Hydraulique

Intitulé

*Prévision des variables
climatologiques aux pas de temps
mensuel et journalier par les réseaux
de neurones et la logique floue*

Chayma SELAMNIA

Sous la direction de : **Mr Nouredine Dechemi** professeur

Présenté et soutenu publiquement le (18/06/2016)

Composition du Jury :

Président	Mr. Maamar NAKIB,	Docteur	ENP
Rapporteur/ Promoteur	Mr. Nouredine DECHEMI,	Professeur	ENP
Examineur	Mr .Salim BENZIADA.	Docteur	ENP

Promotion juin 2016



الحمد والشكر لله رب العالمين

Remerciements :

En préambule, je souhaite rendre grâce à Dieu, le clément et le miséricordieux de m'avoir donné la force et la patience de mener à bien ce modeste travail.

Les remerciements ont et auront toujours une place particulière dans un travail de thèse. Puisqu'il s'agit de l'unique moment où il est possible de rendre, un peu à la manière d'un générique de film, la place qu'est la leur aux personnes ayant tiré de près ou de loin les ficelles de ses humeurs, de sa motivation, et de son bien être professionnel et personnel. J'ose espérer que je n'oublierai personne.

Mes plus profonds, entiers et sincères remerciements iront sans détours à mon encadrant, mon professeur, **Mr Noureddine Dechemi**, pour son implication qui ne s'est jamais tarie durant cette année (bien au contraire) et son enthousiasme légendaire. La tranquille confiance dont vous m'avez témoigné a été un vrai moteur pour m'amener au bout de ce travail : toujours rassurant, toujours compréhensifs et toujours cet incroyable don à trouver des idées pour m'aider à retrouver mon chemin. Merci à vous monsieur pour avoir ressortir toutes mes capacités, mes potentialités et le plus important de me permettre rentrer dans le monde de la recherche et de sentir son paradoxe celui de la joie et la douleur .Milles mercis.

Je tiens également à exprimer ma profonde reconnaissance aux **membres du jury** d'avoir accepté prendre connaissance de mon travail et bien avoir voulu l'évalué.

Un merci infini à mes parents, qui m'ont énormément donne et toujours soutenu dans chacune de mes décisions ; Merci à **mon père** pour m'avoir montré l'importance de l'effort et du travail bien fait ; Merci à **ma mère** pour son amour indéfectible. Sans trait d'esprit, rien de tout cela n'aurait été possible sans vous. Sans oublier **mon frère et sœurs** pour leur bienveillance et leur bonne humeur.

Je voudrais aussi adresser mes remerciements à ma mère Saida et mon père Chérif pour leur aimable gentillesse.

*J'aimerais terminer ces remerciements par la personne ayant partagé tous moment durant l'élaboration de ce travail, Je ne saurai jamais comment te remercier **Amine** pour ta patience, ton soutien, ton aide et ton réconfort. C'était une belle preuve d'amour que tu m'as faite. Du fond du cœur, le plus grand des mercis à toi.*

ملخص

الإدارة المثلى لأي سد تواجه مشكلتين رئيسيتين : من جهة معرفة نتائج تلبية الاحتياجات ليست أكيدة بالمرّة، و من جهة أخرى إذن التنبؤ تنبيهات سواء في الجفاف أو في حالة فيضان. الصعوبة البالغة للتنبؤ بكميات المياه المتدفقة قبل و أثناء فترات الجفاف من خلال هذا البحث نستعرض الإطار العام لتطوير إدارة سد . بالمستقبل الغامض يعتبر أداة قيمة من أجل تحسين إدارة السد وإصدار ، (الشفافية و لاسيما من خلال اقتراح اجراءات استنادا إلى نتائج نماذج التنبؤ بالتغيرات المناخية: دورية (كمية المياه المتبخرة) و التي تم تطويرها وفقا لتقنيتين مختلفتين من تقنيات الذكاء الاصطناعي (كمية التساقط) و متأثرة بتغيرات مختلفة) عشوائية الحجم ألا و هما المنطق الضبابي و شبكات الأعصاب الإصطناعي

كلمات البحث

توقعات، والشبكات العصبية والمنطق الضبابي، والإدارة، الخزان

Abstract:

The optimal management of a reservoir meets two main difficulties: first the very uncertain knowledge of the cost of an objective satisfaction of failure, on the other hand the high uncertainty on forecast rates made before and during the low water. Forecasting uncertain future is a valuable tool for establishing a "good" management and issue alerts in case of drought or flooding.

We review here the general framework for the development of the management of the dam Cheffia, notably by proposing alternatives decision procedures; based on the results of predictive models of cyclical climate variables (the evaporated blades), random (intakes) and influence (the volume), developed at the base of two different techniques of artificial intelligence namely networks neural and fuzzy logic.

Key Words:

Forecast, neural networks, fuzzy logic, management, reservoir.

Résumé:

La gestion optimale d'un réservoir rencontre deux difficultés principales: d'une part la connaissance très incertaine du coût d'une défaillance de satisfaction des objectifs, d'autre part les fortes incertitudes sur les prévisions de débits faites avant et au cours de l'étiage. La prévision en avenir incertain est un outil précieux pour établir une "bonne" gestion et de diffuser des alertes en cas de sécheresse ou inondation.

On réexamine ici le cadre général de la mise au point de la gestion du barrage de la CHEFFIA, en proposant notamment d'autres alternatives de procédures de décision ; basées sur les résultats des modèles de prévision des variables climatologiques cyclique : (les lames évaporées), aléatoire (les apports), et influencé (le volume), élaborés à la base de deux techniques différentes de l'intelligence artificielle à savoir les réseaux de neurones et la logique floue.

Mots clés :

Prévisions, réseaux de neurones, logique floue, gestion, réservoir.

Sommaire :

<i>Liste des tableaux</i>	6
<i>Liste des figures</i>	8
<i>Introduction Générale</i>	11
<i>Chapitre 1 : La modélisation hydrologique</i>	13
1.1 Introduction :.....	14
1.2 Les échelles d'analyse	14
1.2.1 L'échelle spatiale d'analyse :	14
1.2.2 L'échelle temporelle d'analyse (annuelle, mensuelle, journalière et horaire):	15
1.2.3. Classification des modèles :	15
1.2.3.1. Modèles déterministes	15
1.2.3.2. Modèles conceptuels	15
1.2.3.3. Modèles physiques	15
1.2.3.4. Modèles stochastiques	15
1.3. Mise en œuvre des modèles :	15
1.3.1. Calage des modèles :	15
1.3.2. Calibrage :	16
1.3.3. Le calage manuel	16
1.3.4. Le calage automatique	16
1.4. Validation des modèles	16
1.5. Exploitation des modèles	16
1.6. Critères d'évaluation et d'efficacité des modèles :	16
1.6.1. Évaluation graphique :.....	17
1.6.2. Évaluation statistique :	17
1.6.3. Les critères adimensionnels :	17
1.6.4. Les indices d'erreurs :	18
<i>Chapitre 2 : Les réseaux de neurones</i>	19
2.1. Introduction :.....	20
2.2. Historique des réseaux de neurones artificiels :	21
2.3. Le neurone biologique :.....	22
2.4. Le neurone formel (artificiel) :	22
2.5. Le comportement des neurones artificiels :	23
2.6. Structure d'interconnexion des réseaux de neurones :	24
2.7. Apprentissage des réseaux de neurones artificiels :	26
2.7.1. La loi de Hebb, un exemple d'apprentissage non supervisé :	27

2.7.2. Le Perceptron et l'Adaline:	29
2.7.3. Le perceptron multicouche:	30
2.9. La Backpropagation :	31
2.9.1. Méthode d'optimisation par la Backpropagation :	31
2.9.2.. Algorithme de Newton :	31
2.9.3. Algorithme de Levenberg-Maquardt :	32
2.10. Conclusion :	32
Chapitre 3 : La logique floue	33
3.1. Introduction :	34
3.2. Historique de la logique floue :	34
3.3. Rappel sur les ensembles classiques :	35
3.4. Les sous-ensembles flous : concepts et définitions	36
3.5. Éléments caractéristiques des sous-ensembles flous :	38
3.6. Principe fondamentale de la logique floue :	40
3.7. Les variables linguistiques :	40
3.8. Les opérateurs flous :	41
3.9. Le raisonnement en logique floue :	42
3.10. Conclusion :	42
3.11. La fuzzification :	43
3.12. La défuzzification :	44
3.13. Association d'un modèle flou neuronal : l'approche neurofloue	45
Chapitre 4 : Élaborations des modèles de prévisions	47
4.1. Introduction.....	48
4.2. L'architecture	48
4.3. Le perceptron multicouches :	49
4.4. L'apprentissage :	49
4.5. Programmation du modèle neuronale:.....	49
4.5.1. Critères de performance :	49
4.5.2.. Résultats et interprétations :	49
4.6. Prévisions mensuelles par les réseaux de neurones :	50
4.6.1. Prévision mensuelle des volumes:	50
4.6.2. Prévision mensuelle des lames évaporées.....	52
4.6.3. Prévisions mensuelles des apports	54
4.7. Prévisions mensuelles par la logique floue	55
4.7.1. Prévision mensuelle des lames évaporées.....	55

4.7.2. Prévision mensuelle des volumes:	56
4.7.3. Prévisions mensuelles des apports	57
4.8. Les prévisions journalières par les réseaux de neurones :.....	58
4.8.1. Prévisions journalières des apports :	58
4.8.2. Prévisions journalières des Volumes:.....	60
4.8.3. Prévisions journalières des lames évaporées :.....	61
4.9. Prévisions journalières par la logique floue :	63
4.9.1. Prévisions des apports :.....	63
4.9.2. Prévisions journalières des lames évaporées :.....	64
4.9.3. Prévisions journalières des volumes	65
4.10. Prévision avec et sans correction des volumes	66
4.10.1. Prévision sans correction:	66
4.10.2. Prévision avec correction:	67
<i>Conclusion Générale</i>	69
<i>Bibliographie</i>	71
<i>Annexes</i>	74

Liste des tableaux :

Table 2.1. La loi de Hebb	28
Table 3.1 : Les opérateurs de la logique floue	41
Table 4.1: Résultats de la prévision mensuelle des volumes avec un décalage (calage)	50
Table 4.2: Résultats de la prévision mensuelle des volumes avec un décalage (validation)	51
Table 4.3 : Résultats de la prévision mensuelle des volumes avec trois décalages (calage)	51
Table 4.4: Résultats de la prévision mensuelle des volumes avec un décalage (validation)	52
Table 4.7 : Résultats de la prévision mensuelle des lames évaporées avec un décalage (calage)	52
Table 4.8 : Résultats de la prévision mensuelle des lames évaporées avec un décalage (validation)..	52
Table 4.9 : Résultats de la prévision des lames évaporées mensuelles avec trois décalages (calage) .	53
Table 4.10: Résultats de la prévision mensuelle des lames évaporées avec trois décalages (validation)	53
Table 4.11 : Résultats de la prévision mensuelle des apports avec un décalage (calage)	54
Table 4.12 : Résultats de la prévision mensuelle des apports avec un décalage (validation)	54
Table 4.13 : Résultats de la prévision mensuelle des apports avec trois décalages (calage)	54
Table 4.14: Résultats de la prévision mensuelle des apports avec un décalage (validation)	55
Table 4.15 : Résultats de prévisions par la logique floue des lames évaporées avec un décalage (calage)	55
Table 4.16 : Résultats de prévision par la logique floue des lames évaporées avec un décalage (validation)	55
Table 4.17: Résultats de prévision par la logique floue des Lames évaporées avec trois décalages (calage)	56
Table 4.18 : Résultats de prévision par la logique floue des lames évaporées avec trois décalages (validation)	56
Table 4.19 : Résultats de prévision par la logique floue des volumes avec un décalage (calage)	56
Table 4.20 : Résultats de prévision par la logique floue des volumes avec un (1) décalage (validation)	56
Table 4.21 : Résultats de prévision par la logique floue des volumes avec trois décalages (calage)...	57
Table 4.22: Résultats de prévision par la logique floue des volumes avec trois décalages (validation)57	57
Table 4.23 : Résultats de prévision des apports par la logique floue avec un décalage (calage)	57
Table 4.24 : Résultats de prévision des apports par la logique floue avec un décalage (validation)....	58
Table 4.25 : Résultats de prévision des apports par la logique floue avec trois décalages (calage).....	58
Table 4.26 : Résultats de prévision des apports par la logique floue avec trois décalages (calage).....	58
Table 4.27 : Résultats de prévision des apports journaliers avec un décalage (calage)	59
Table 4.28 : Résultats de prévision des apports journaliers avec un décalage (validation)	59
Table 4.29 : Résultats de prévision des apports journaliers avec trois décalages (calage).....	59
Table 4.30 : Résultats de prévision des apports journaliers avec trois décalages (validation)	60
Table 4.31 : Résultats de prévision des volumes journaliers avec un décalage (validation)	60
Table 4.32: Résultats de prévision des volumes journaliers avec trois décalages (calage)	61
Table 4.33: Résultats de prévision des volumes journaliers avec un décalage (validation)	61
Table 4.34: Résultats de prévision des lames évaporées journalières avec un décalage (calage)	62
Table 4.35 : Résultats de prévision des lames évaporées journalières avec un décalage (calage)	62
Table 4.36 : Résultats de prévision des lames évaporées journalières avec trois décalages (calage)..	62
Table 4.37 : Résultats de prévision des lames évaporées journalières avec trois décalages (calage)..	63
Table 4.38 : Résultats de prévisions la logique floue des apports journaliers avec un décalage (calage)	63

Table 4.39 : Résultats de prévision des apports journaliers par la logique floue avec un décalage (validation)	63
Table 4.40 : Résultats de prévision apports journaliers par la logique floue avec trois décalages (calage)	64
Table 4.41 : Résultats de prévision des apports journaliers par la logique floue avec trois décalages (validation)	64
Table 4.42: Résultats de prévisions la logique floue des lames évaporées avec un décalage (calage)	64
Table 4.43 : Résultats de prévision des lames évaporées journalières par la logique floue avec un décalage (calage)	64
Table 4.44 : Résultats de prévision des lames évaporées journalières par la logique floue avec trois décalages (calage)	65
Table 4.45 : Résultats de prévision des volumes journaliers par la logique floue avec trois décalages (validation)	65
Table 4.46 : Résultats de prévision des volumes journaliers par la logique floue avec un décalage (calage)	65
Table 4.47 : Résultats de prévision des volumes journaliers par la logique floue avec un décalage (validation)	65
Table 4.48 : Résultats de prévision des volumes journaliers par la logique floue avec trois décalages (calage)	66
Table 4.49 : Résultats de prévision des volumes journaliers par la logique floue avec un décalage (validation)	66
Table 4.50 : Résultats d'une Prévision sans correction des volumes.....	67
Table IV.51:Résultats d'une prévision avec correction des volumes.....	68

Liste des figures :

Figure 2.1 : Représentation simplifié d'un neurone biologique.....	22
Figure 2.2 : Schéma représentatif d'u neurone artificiel	23
Figure 2.3 : Définition d'un réseau multicouches	24
Figure 2.4 : Réseau à connexions locales	25
Figure 2.5 : Réseau à connexions récurrentes.	25
Figure 2.6 : Réseau à connexions complète	26
Figure 2.6 : Connexion entre deux neurones	27
Figure 2.7 : L'ADALINE de Widrow-Hoff	30
Figure 2.8 : Perceptron multicouches	30
Figure 3.1 : Représentation graphique de l'ensemble {1 ; 5 ; 6 ; 7 ; 10}.....	35
Figure 3.2: Fonction d'appartenance caractérisant le sous-ensemble 'bon' de la..... qualité du service	37
Figure 3 .3: Exemple de fonctions d'appartenance.....	38
Figure 3. 5: Deux représentations possibles classique et par logique floue des personnes selon leur âge	40
Figure 3.6 : Exemple de variable linguistique 'qualité de service	41
Figure 3.7: Composantes d'un système flou	43
Figure 3.8: Défuzzification par le centre de gravité	44
Figure 3.9 : Modèle adaptatif ANFIS	46

Introduction Générale

La croissance démographique, l'urbanisation et les progrès réalisés en termes d'industrialisation se combinent pour créer une demande en eau toujours plus importante. Les écosystèmes, milieux producteurs et régénérateurs de cette ressource sont menacés, pollués et détruits, Les déficits et les excédents de précipitations (sécheresse-inondations), les fortes pluies d'orage dans les régions très urbanisées ont, sur les activités humaines, des conséquences souvent dommageables.

Face à ce constat alarmant des prévisions fiables de certaines variables climatologiques à longue échéance (quelques semaines à quelques mois à l'avance) peuvent améliorer la gestion des ressources en eau et de ce fait l'économie de la société et les conditions de vie aquatique ;

Les prévisions hydrologiques ont pour objectif de permettre une planification plus éclairée des interventions, tant pour des situations de crue ou d'étiage que dans des conditions hydrologiques plus courantes.

Le progrès dans la modélisation hydrologique ainsi que dans l'optimisation de l'exploitation des barrages-réservoirs a pu fournir aux gestionnaires une grande variété de modèles mathématiques et d'outils informatiques pour les aider dans leur tâche difficile (Labadie, 2004), et beaucoup de travaux ont été consacrés pour améliorer la gestion des barrages en ces périodes critiques. Au paravent les chercheurs ont l'habitude d'utiliser des modèles d'optimisation déterministes pour produire des prévisions hydrologiques (Valdes et Marco, 1995) et (Giron et all, 2000), mais ce type de modèle ne peut pas tenir en compte l'incertitude dans les paramètres du modèle ou les futurs apports.

Avec l'évolution de l'informatique, plusieurs techniques mathématiques alternatives ont été développées, pour la protection des barrages contre les inondations, et des approches basées sur les techniques d'intelligence artificielle semblent être prometteuses pour la gestion des réservoirs. Karaboga et all (2004). Ces techniques sont plus flexibles que l'optimisation mathématique classique, et dans beaucoup de cas, elles produisent des prévisions plus précises et de meilleures recommandations dans la gestion. Mediero (2007) ; Les principaux modèles sont : les réseaux de neurones artificiels, la logique floue, les algorithmes génétiques, et les systèmes neuro-flou.

Notre objectif est donc d'élaborer des modèles basés sur les réseaux de neurones et la logique floue permettant d'émettre pour les n mois à venir une prévision au pas de temps journalier ainsi que mensuel des trois variables : les apports, les lames évaporées et les volumes, afin d'adapter l'opération du barrage de la Cheffia face aux aléas climatologiques et hydrologiques, dans le but de respecter les plans de gestion.

Chapitre 1 : La modélisation

hydrologique

1. Introduction :

La modélisation est une traduction de l'information réelle en information mathématique, puis la traduction des résultats mathématiques en prédiction des opérations réelles.

La modélisation dans le domaine de la recherche, consiste en la construction du modèle et la définition des règles théoriques le régissant. Quant à la modélisation opérationnelle, c'est la vérification des principes de base de construction du modèle par rapport aux objectifs et finalités fixés.

Le développement du modèle repose généralement sur trois éléments:

1. Le système observé et sa discrétisation spatiale et temporelle, qui en définissent l'objet et ses limites (spatiales ou temporelles). La connaissance du système est conditionnée par la mesure de ses caractéristiques et l'acquisition de données sur les flux, les stocks et les transformations de phase. Dans notre cas, le système est typiquement le bassin versant défini plus loin;

L'objectif de modélisation, pour lequel le modèle est développé. Le modèle est construit pour répondre à des questions et peut ainsi, au-delà d'un outil de représentation, être également un instrument de connaissance. L'étude du cycle de l'eau est l'objectif général de l'hydrologie. Nous nous intéresserons plus particulièrement dans ce travail de recherche à la représentation de la transformation de la pluie en débit;

Le choix d'une formulation de la réalité. Après la définition des deux points précédents.

Parce qu'une très grande complexité caractérise les processus impliqués dans le cycle de l'eau, les hydrologues conviennent aujourd'hui de leurs connaissances partielles. En effet, la représentation de tout ou partie de ce cycle (démarche de modélisation), à des fins variées, se heurte à la difficulté d'appréhension, de description ou de compréhension des phénomènes et des systèmes étudiés. Par conséquent, les représentations de ces systèmes, c'est-à-dire les modèles, sont nécessairement simplificatrices, réductrices de la complexité naturelle, et donc grossièrement inexactes. Le modélisateur opère des choix de représentation du système en ne retenant que les aspects qui lui semblent les plus pertinents, pour tendre vers une solution la moins inexacte possible.

S'il est souvent possible de construire un très grand nombre de modèles sur une réalité particulière, les points 1 et 2 fournissent des balises généralement capables de restreindre les choix du modélisateur (Thom, 1979, cité par Nascimento, 1995).

2. Les échelles d'analyse

2.1. L'échelle spatiale d'analyse :

Définit l'objet et la limite de la modélisation, et on distingue :

- La modélisation globale qui étudie le bassin versant comme une seule entité aux propriétés uniformes, ayant une réponse globale et une paramétrisation unique afin d'en tirer les caractéristiques et en comprendre le fonctionnement.

2.2. L'échelle temporelle d'analyse (annuelle, mensuelle, journalière et horaire):

Dépend de l'objectif scientifique fixé pour la modélisation. Plus l'échelle temporelle diminue et plus le problème se complique en raison des états antérieurs du bassin.

2.3. Classification des modèles :

2.3.1. Modèles déterministes

Modèle qui associe à chaque jeu de variables de forçage, de variables d'état et de paramètres, une valeur réalisée unique des variables de sortie.

2.3.2. Modèles conceptuels

Modèles basés sur le concept du bilan hydrologique. Ces modèles sont représentés par des formules empiriques. La cohérence hydrologique est fondamentale pour ce type de modèles. Le bassin versant est considéré comme un assemblage de réservoirs interconnectés représentant des niveaux de stockage suivant une dimension verticale.

Il existe plusieurs modèles de ce type : les modèles GR, CREC, Thorentwaite, ABCD, etc.

2.3.3. Modèles physiques

Basés sur les lois de la physique et ne comportant idéalement aucun paramètre. L'étude ici n'est pas globale mais discrétisée (maillée), et vu la complexité des processus et du milieu étudié (hétérogénéité), ce type de modèles reste très fastidieux, complexe et délicat à l'exploitation.

2.3.4. Modèles stochastiques

Basés sur un processus aléatoire qui relie les données d'entrée aux données de sortie. Ils reposent essentiellement sur les données observées et les concepts statistiques. Ils ne suivent aucune loi physique ou hydrologique, d'où leur appellation "Boite noire". Les principaux modèles « Boite noire » sont : les réseaux de neurones artificiels, la logique floue, les algorithmes génétiques, et les systèmes neuro-flou.

3. Mise en œuvre des modèles :

3.1. Calage des modèles :

Le calage consiste en l'ajustement numérique des entrées afin de reproduire les observations. Ce processus d'ajustement des variables d'entrée et des paramètres du modèle est établi dans le but de reproduire certains aspects de la réalité en comparant des variables de sortie à des données observées.

Il s'agit de l'une des étapes les plus consommatrices en temps de travail lors de la construction du modèle. Elle est fondamentale dans la mesure où elle détermine la qualité du modèle (Setra, 2010).

3.2. Calibrage :

Processus de calage, consiste à déterminer les paramètres d'un modèle à partir d'un jeu de données observées (Setra, 2010).

3.3. Le calage manuel

La détermination d'un jeu de paramètres par tâtonnement permettant d'obtenir un optimum, qui est atteint lorsque l'erreur est minimisée entre les données simulées et observées, l'inconvénient du calage manuel est qu'il est fastidieux, et nécessite un temps relativement long.

3.4. Le calage automatique

Dans ce cas, les données simulées sont ajustées aux données observées, au moyen d'un programme informatique, cet algorithme est plus avantageux que le calage manuel, mais présente l'inconvénient d'être influencé par certaines contraintes, qu'on résume en (Benkaci, 2006) :

- L'interdépendance des paramètres (autocorrélation) : le changement de la valeur d'un paramètre peut être compensé par la modification d'autres paramètres, ce qui entraîne des solutions équivalentes en terme de valeur du critère d'optimisation;
- La faible sensibilité de la fonction objectif;
- La présence d'optima locaux, ou également la présence de plusieurs zones de convergences.

La qualité de l'estimation des paramètres d'un modèle dépend notamment de la puissance et de la robustesse de l'algorithme utilisé.

4. Validation des modèles

Souvent confondue avec l'étape de calage, la validation consiste à s'assurer que le modèle réagit correctement à des variations des données d'entrée et qu'il est apte à fonctionner en projection. On utilise pour cette étape des données qui n'ont pas servi au calage du modèle.

Elle permet également de vérifier la sensibilité du système aux changements à analyser afin de s'assurer que les écarts entre résultats obtenus et observés sont raisonnables.

5. Exploitation des modèles

Une fois le modèle calé et validé, il peut être utilisé en projection, pour différents usages, en s'assurant à chaque exploitation qu'il reste pertinent pour répondre aux problèmes posés.

6. Critères d'évaluation et d'efficacité des modèles :

L'utilisation des modèles pluie-débit doit être très soignée vu leurs domaines d'utilisation. Il n'existe pas un bon modèle, mais juste un modèle meilleur qu'un autre.

6.1. Évaluation graphique :

L'évaluation graphique nous donne un premier aperçu sur la tendance de la variable simulée par rapport à la variable observée. Elle permet de détecter certaines anomalies mal décelables par les critères numériques classiques. Après calage des modèles, elle nous indique si l'évolution de la variable simulée suit l'évolution de la variable observée. Autrement dit, elle nous indique si le modèle reproduit bien les événements.

6.2. Évaluation statistique :

Afin de juger l'efficacité des modèles, on présentera les critères appliqués dans le domaine de la modélisation hydrologique. Une étude comparative du degré de représentation de ces critères sera établie par la suite afin de juger de la pertinence des critères suivant notre cas d'étude.

Les critères globaux d'appréciation des performances des modèles de simulation des débits diffèrent sur trois points essentiels: la forme analytique de l'erreur (quadratique, absolue, cumulée...), la forme de la variable cible (transformée ou non) et le choix d'un modèle de référence (pour avoir une mesure relative) (Perrin, 2000).

La régression standard

Évaluent la force de la relation linéaire existant entre les observations et les prédictions.

Coefficient de détermination

Le coefficient de détermination r^2 est défini comme étant le carré du coefficient de corrélation, d'après Bravais-Pearson :

$$r^2 = \left[\frac{\sum_{i=1}^n (Qo.i - Qo.moy)(Qs.i - Qs.moy)}{\sqrt{\sum_{i=1}^n (Qo.i - Qo.moy)^2} \sqrt{\sum_{i=1}^n (Qs.i - Qs.moy)^2}} \right]^2$$

Le r^2 estime la dispersion (distribution) entre les valeurs simulées et observées, en déterminant la proportion avec laquelle la dispersion estimée explique la dispersion observée.

Il varie de 0 (pas de corrélation) à 1 (la distribution de la prédiction est égale à celle de l'observation).

6.3. Les critères adimensionnels :

Coefficient de NASH-SUTCLIEFFE :

Proposé par Nash et Sutcliffe en 1970, c'est un critère qui évalue le pouvoir prédictif des modèles hydrologiques. Il est défini comme étant « un » moins la somme des carrés des erreurs normalisées par la variance des valeurs observées durant la période d'étude :

$$NSE = 1 - \frac{\sum_{i=1}^n (Q_o - Q_s)^2}{\sum_{i=1}^n (Q_o - Q_o.moy)^2}$$

Le NSE varie entre "-∞" et "1.0". Un NSE de 1.0 représente une simulation parfaite, et un NSE indique que la moyenne des observations représente un meilleur indice que la valeur simulée par le modèle, ce qui représente une performance inacceptable.

6.4. Les indices d'erreurs :

MAE, MSE, RMSE :

L'erreur moyenne absolue, l'erreur quadratique moyenne, et racine de l'erreur quadratique moyenne, sont les indices d'erreurs les plus utilisés. Ils sont intéressants car ils indiquent l'erreur dans les unités des variables étudiées. Une valeur de « 0 » de ces indices indique que le modèle simule bien la variable étudiée.

$$MAE = \frac{1}{n} \sum_{i=1}^n |Q_o - Q_s|$$

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

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Q_o - Q_s)^2}$$

Percent Bias (PBIAS)

Mesure la tendance moyenne des valeurs simulées (plus grande ou plus petite) par rapport aux valeurs observées (GUPTA et al., 1999), donc estime la sous/surestimation de la simulation. Sa valeur optimale est de « 0 ». Des valeurs positives de ce critère indiquent qu'il y a sous-estimation du biais et des valeurs négatives indiquent qu'il y a surestimation du biais.

$$PBIAS = \frac{\sum_{i=1}^n (Q_o - Q_s)}{\sum_{i=1}^n Q_o}$$

Chapitre 2 : Les réseaux de

neurones

1. Introduction :

L'informatique est la science du traitement automatique de l'information. Son développement est souvent confondu avec celui des machines de traitement : les ordinateurs. Depuis les débuts jusqu'à aujourd'hui, les ordinateurs sont devenus de plus en plus puissants.

Cependant, cette augmentation de puissance ne permet pas de toujours résoudre les problèmes d'une application informatique dans un domaine particulier. Ceci revient à ce que certaines opérations ne trouvent pas de solutions, et se heurtent à des difficultés, tels que le grand nombre de données, leur nature et leur variabilité, ainsi que la nécessité des calculs en temps réel.

Les programmes informatiques conçus avaient pour objectif de résoudre la complexité des calculs numériques, et du temps de leur exécution. Ainsi on voit naître la première approche du traitement informatique, l'approche algorithmique, qui nécessite l'écriture (avant la transcription dans un quelconque langage de programmation) du processus à suivre pour résoudre le problème. Lorsque le problème est complexe, ce peut être une étape coûteuse ou impossible. D'autre part, les ordinateurs sont des machines complètement logiques (et même binaires) qui suivent à la lettre chacune des instructions du programme. C'est un avantage lorsque tous les cas ont été prévus à l'avance par l'algorithmicien. Cependant, la nature très complexe de certains phénomènes (physiques ou autres) ne permet pas à cette technique d'aboutir devant une situation imprévue ou inadaptée.

Pour faire face à ces problèmes ; les chercheurs ont recours à une deuxième approche qui est celle de l'intelligence artificielle, en vue d'améliorer les capacités de résolution d'un système informatique, est de lui inclure des connaissances sur le domaine qu'il doit gérer.

Donc Comment l'homme fait-il pour raisonner, parler, calculer, apprendre, ...? Comment s'y prendre pour créer une ou de l'intelligence artificielle ? Deux types d'approches ont été essentiellement explorés :

- procéder d'abord à l'analyse logique des tâches relevant de la cognition humaine et tenter de les reconstituer par programme. C'est cette approche qui a été privilégiée par l'Intelligence Artificielle et la psychologie cognitive classique, Cette démarche est étiquetée sous le nom de *cognitivism*.
- puisque la pensée est produite par le cerveau ou en est une propriété, commencer par étudier comment celui-ci fonctionne. C'est cette approche qui a conduit à l'étude de réseaux de neurones formels. On désigne par *connexionnisme* la démarche consistant à vouloir rendre compte de la cognition humaine par des réseaux de neurones.

La seconde approche a donc menée à la définition et l'étude de réseaux de neurones formels qui sont des réseaux complexes d'unités de calcul élémentaire interconnectées.

Les réseaux de neurones artificiels ont été développés avec pour objectifs principaux d'une part la modélisation et compréhension du fonctionnement du cerveau et d'autre part pour réaliser des architectures ou des algorithmes d'intelligence artificielle.

2. Historique des réseaux de neurones artificiels :

Les travaux de recherches avaient pour objectif de s'approcher le plus possible du fonctionnement du cerveau humain, afin de bénéficier de plusieurs de ses fascinantes caractéristiques.

D'où une analogie avec le système nerveux humain a fait naître le principe de réseaux de neurones en 1940. et en 1943 deux chercheurs Mc Culloch et Pitts ont inventé le premier "neurone artificiel", qui correspondait à un produit d'un vecteur d'entrées, et un vecteur poids suivi d'un élément de seuillage. Ceux sont les premiers à montrer que des réseaux de neurones formels simples peuvent réaliser des fonctions logiques, arithmétiques et symboliques complexes (tout au moins au niveau théorique). Cette modélisation du comportement du neurone humain a ouvert la voie à une nouvelle recherche, tel que le physiologiste américain D. Hebb Expliqua le conditionnement chez l'animal par les propriétés des neurones humains et proposa une théorie fondamentale de l'apprentissage en 1949 Le concept d'apprentissage par adaptation des poids synaptiques des réseaux a vu le jour, puisque Minsky (1951) construit ce qui fut le premier modèle réel de réseaux de neurones baptisé Snarl. En 1957, F. Rosenblatt développe le modèle du Perceptron. Il construit le premier neuroordinateur 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. Trois ans plus tard, Widrow et Hoff (1960) développèrent un nouveau modèle neuronal, l'Adaline (Adaptative Linear Neurone). Dans sa structure, le modèle ressemble au Perceptron, cependant la loi d'apprentissage est différente. Celle-ci est à l'origine de l'algorithme de rétro propagation de gradient très utilisé aujourd'hui avec les Perceptrons multicouches. En 1969, Minsky et Papert publièrent un ouvrage critique qui met en exergue les limitations théoriques du Perceptron, concernant l'impossibilité de traiter des problèmes non linéaires.

À partir des années quatre-vingt, avec le développement des mathématiques et de la physique, les recherches sur les réseaux de neurones s'intensifient, ainsi le physicien Hopfield (1982), puis Grossberg (1987), ont pu résoudre pour la première fois le problème de stabilité des réseaux dynamiques.

En 1986, une équipe de chercheurs dirigée par Rumelhart, a pu concevoir un nouvel algorithme d'apprentissage, constitué d'un ensemble de couches (appelé Perceptron multicouches). Sa principale caractéristique est l'approximation de fonction non-linéaire, par la rétro propagation de gradient (ou Back propagation). Cette découverte, a permis de réaliser une fonction non linéaire d'entrée/sortie, et a ouvert une nouvelle ère pour les réseaux de neurones, mais ils ne résolvaient pas tous les problèmes. L'apprentissage, notamment, était difficile et, là aussi, les limites étaient rapidement atteintes.

En 2006, Geoffrey Hinton revint avec encore un nouveau modèle de réseau neuronal, celui qui est justement appelé "deep learning". Cette fois, les différentes couches de neurones effectuent un apprentissage hiérarchique. Autrement dit, elles cherchent d'abord à catégoriser les éléments les plus simples de l'information avant de passer aux plus compliqués.

Actuellement, les réseaux de neurones disposent de fondements théoriques solides et constituent donc un ensemble d'opérateurs linéaires et non-linéaires, qui permettent grâce à leur adaptabilité une large utilisation dans différentes disciplines scientifiques.

En hydrologie connaissant la non-linéarité de la relation pluie-débit, certains auteurs ont exploité les réseaux de neurones artificiels dans la modélisation pluie débit à différentes échelles (French et al. 1994, Shamseldin, 1996, Benkaci, 2006).

Donc Comment l'homme fait-il pour raisonner, parler, calculer, apprendre, ...? C'est ce que nous amène d'abord à définir le neurone biologique.

3. Le neurone biologique :

Le neurone est une cellule spécialisée dans le traitement et la transmission de l'information, Le neurone est une cellule composée d'un corps cellulaire et d'un noyau. Le corps cellulaire se ramifie pour former ce que l'on nomme les dendrites. Celles-ci sont parfois si nombreuses que l'on parle alors de chevelure dendritique ou d'arborisation dendritique. C'est par les dendrites que l'information est acheminée de l'extérieur vers le soma, corps du neurone.

L'information traitée par le neurone chemine ensuite le long de l'axone (unique) pour être transmise aux autres neurones. La transmission entre deux neurones n'est pas directe. En fait, il existe un espace intercellulaire de quelques dizaines d'Angströms (10^{-9} m) entre l'axone du neurone afférent et les dendrites (on dit *une* dendrite) du neurone efférent. La jonction entre deux neurones est appelée la synapse (Figure.1).

La notion de *synapse* explique la transmission des signaux entre un axone et une dendrite. Au niveau de la jonction (c'est à dire de la synapse), il existe un espace vide à travers lequel le signal électrique ne peut pas se propager. La transmission se fait alors par l'intermédiaire de substances chimiques, les *neuromédiateurs*. Quand un signal arrive au niveau de la synapse, il provoque l'émission de neuromédiateurs qui vont se fixer sur des récepteurs de l'autre côté de l'espace inter-synaptique. Quand suffisamment de molécules se sont fixées, un signal électrique est émis de l'autre côté et on a donc une transmission. En fait, suivant le type de la synapse, l'activité d'un neurone peut renforcer ou diminuer l'activité de ces voisins. On parle ainsi de synapse excitatrice ou inhibitrice.

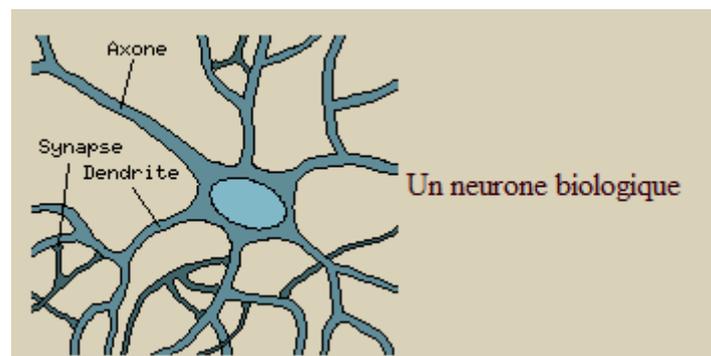


Figure 2.1 : Représentation simplifié d'un neurone biologique

4. Le neurone formel (artificiel) :

Un neurone artificiel est une unité de traitement qui dispose de n entrées $\{x_i\}_{i=1,\dots,n}$ et d'une sortie y , qui se ramifie ensuite pour alimenter un nombre variable de neurones aval. Cette sortie correspond à la transformation par une fonction d'une somme pondérée des entrées, à chacune de ces entrées est associé un poids 'w' abréviation de weight (poids en anglais) représentatif de la force de la connexion.

$$y = f(v) \quad \text{avec} \quad v = w_0 + \sum_{i=1}^n w_i x_i$$

Les quantités $\{w_i\}_{i=0,\dots,n}$ sont les *poids* du neurone. La fonction f est appelée *fonction d'activation* ou fonction de transfert du neurone. Les fonctions les plus communément utilisées sont la fonction échelon unité ou Heaviside, la fonction signe, la fonction linéaire ou semi-linéaire, la fonction tangente hyperbolique, ou la fonction sigmoïde.

Le principe de fonctionnement d'un neurone artificiel peut être résumé par le schéma suivant :

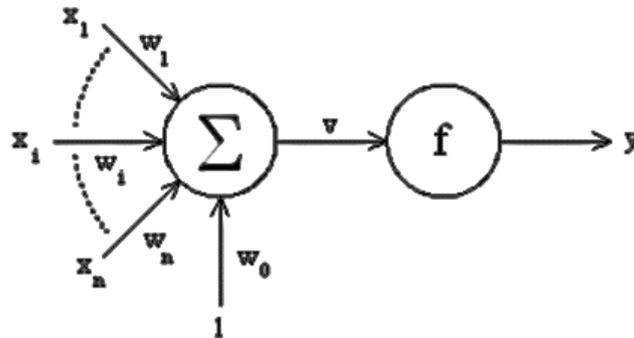


Figure 2.2 : Schéma représentatif d'un neurone artificiel

Un neurone est donc une fonction algébrique, non-linéaire paramétrée, à valeurs bornées (Dreyfus et al. 2004).

5. Le comportement des neurones artificiels :

On distingue deux phases. La première est habituellement le calcul de la somme pondérée des entrées (a) selon l'expression suivante : $a = \sum (W_i \cdot e_i)$

On compare ensuite la somme obtenue à un seuil, et on déduit de la comparaison la sortie du neurone. Cette sortie sera par exemple égale à 1, si la somme est supérieure au seuil et à 0 dans le cas contraire. Plus formellement encore, il suffit pour obtenir ce comportement de soustraire le seuil considéré à la somme des entrées, et de faire passer le résultat par la *fonction de transfert* du

neurone ; Parmi les fonctions les plus utilisées, on retrouve la fonction « seuil » (Hard limit) la fonction linéaire, et la fonction sigmoïde.

On constate tout d'abord que le modèle biologique fait intervenir une notion temporelle qui est difficile à intégrer dans un modèle simple. On oublie donc cette notion et de ce fait on remplace l'intégration temporelle par une simple sommation des signaux arrivant au neurone (ces signaux sont communément appelés les *entrées* du neurone). De plus la plupart des fonctions de transfert sont continue, offrant une infinité de valeurs possibles comprises dans l'intervalle $[0, +1]$ (ou $[-1, +1]$).

6. Structure d'interconnexion des réseaux de neurones :

Un neurone réalise simplement une fonction non linéaire, paramétrée, de ses variables d'entrée. L'intérêt des neurones réside dans les propriétés qui résultent de leur association en réseaux, c'est-à-dire de la *composition* des fonctions non linéaires réalisées par chacun des neurones. Les connexions entre les neurones qui composent le réseau décrivent la topologie du modèle. Selon la structure d'interconnexion de ces réseaux on distingue :

Réseau multicouches: les neurones sont arrangés par couches. Il n'y a pas de connexion entre neurones d'une même couche et les connexions ne se font qu'avec les neurones des couches avales (Figure. 3). Ceci nous permet d'introduire la notion de sens de parcours de l'information (de l'activation) au sein d'un réseau dont l'information à l'intérieur du réseau est sans retour en arrière (Feed-back).

On appelle couche d'entrée l'ensemble des neurones d'entrée, couche de sortie l'ensemble des neurones de sortie. Les couches intermédiaires n'ayant aucun contact avec l'extérieur sont appelés couches cachées.

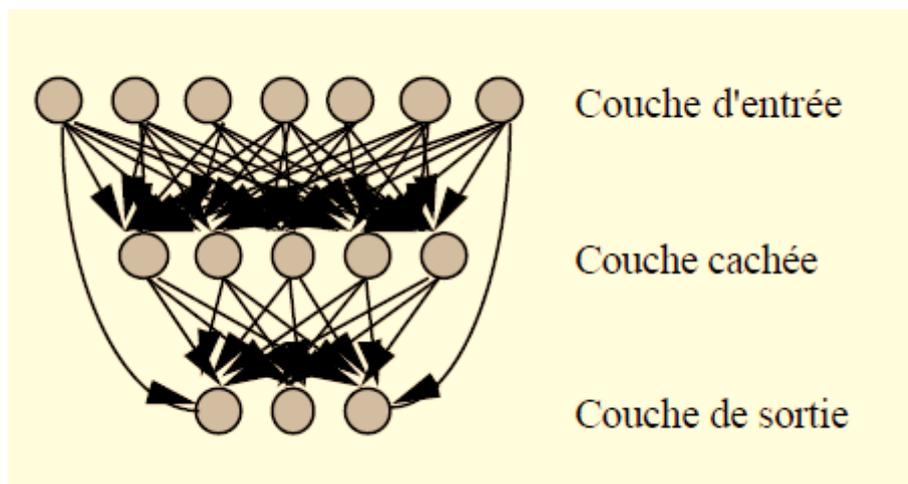


Figure 2.3 : Définition d'un réseau multicouches

Réseau à connexions locales : Chaque neurone de cette structure entretient des relations avec un nombre réduit et localisé de neurones de la couche avale (Figure. 4). Les connexions sont donc moins nombreuses que dans le cas d'un réseau multicouche classique.

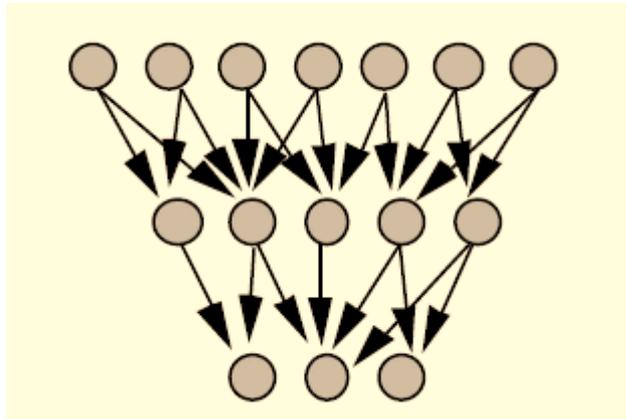


Figure 2.4 : Réseau à connexions locales

Réseau à connexions récurrentes : les connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche. Ces connexions sont le plus souvent locales (Figure II.5).

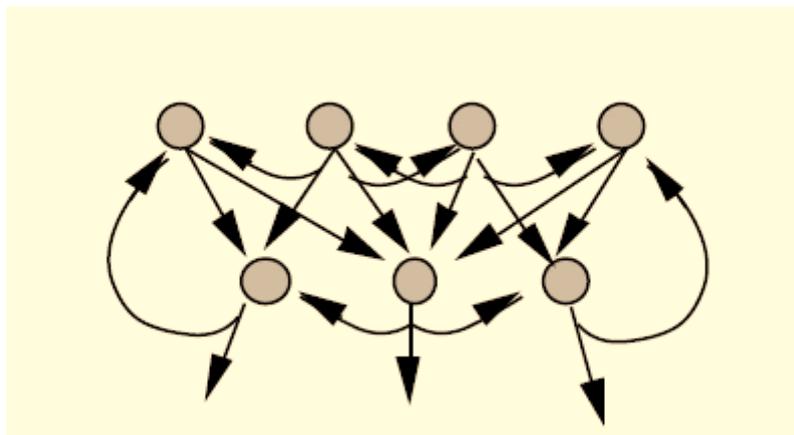


Figure 2.5 : Réseau à connexions récurrentes.

Réseau à connexion complète : c'est la structure d'interconnexion la plus générale (Figure. 6). Chaque neurone est connecté à tous les neurones du réseau (et à lui-même) ; Ces réseaux disposent d'un graphe de connexions *cyclique* : lorsqu'on se déplace dans le réseau en suivant le sens des connexions, il est possible de trouver au moins un chemin qui revient à son point de départ (un tel chemin est désigné sous le terme de « Cycle »). La sortie d'un neurone du réseau peut donc être

fonction d'elle-même ; cela n'est évidemment concevable que si la notion de *temps* est explicitement prise en considération.

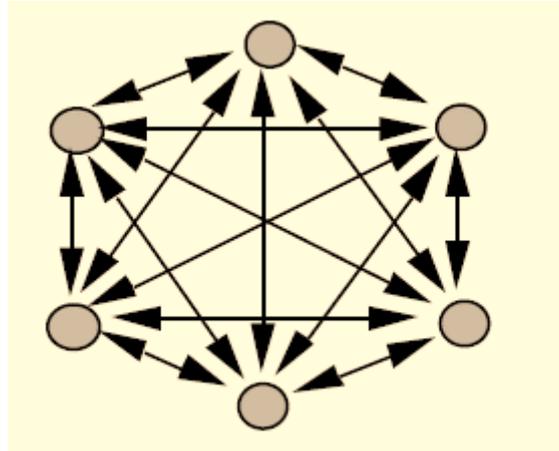


Figure 2.6 : Réseau à connexions complète

7. Apprentissage des réseaux de neurones artificiels :

Dans le cas d'un cerveau humain, l'utilisation correcte des neurones nécessite au moins une phase d'apprentissage qui permettra d'avoir le fonctionnement escompté. Cet apprentissage consiste à jouer sur les synapses qui relient les neurones. De la même manière, les réseaux de neurones artificiels ont besoin d'apprendre.

L'apprentissage est vraisemblablement la propriété la plus intéressante des réseaux neuronaux. Elle ne concerne cependant pas tous les modèles, mais les plus utilisés.

Ce qui nous amène à la définition d'apprentissage des réseaux de neurones artificiels :

Définition :

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré ; pour cela il faudra agir sur les synapses, en réglant les valeurs des poids de chaque connexion entre neurones ainsi que sur les seuils (ou biais).

Il existe plusieurs règles d'apprentissage qui peuvent être regroupées en trois grandes catégories (Coulibaly et al, 1999, Dreyfus et al. 2004) :

Avant d'entamer ces trois catégories il nous faut d'abord définir :

La Base d'apprentissage : exemples représentatifs du comportement ou de la fonction à modéliser. Ces exemples sont sous la forme de couples (entrée ; sortie) connus.

Base d'essai : pour une entrée quelconque (bruitée ou incomplète), calculer la sortie. On peut alors évaluer la performance du réseau.

Donc :

L'apprentissage supervisé : les coefficients synaptiques sont évalués en minimisant l'erreur (entre sortie souhaitée et sortie obtenue) sur une base d'apprentissage.

L'apprentissage non supervisé : Dans l'apprentissage *non supervisé*, il n'y a pas de notion de sortie désirée, on dispose seulement d'un nombre fini de données d'apprentissage, constituées "d'entrées", on ne dispose pas de base d'apprentissage. Remarquons cependant que les modèles à apprentissage non supervisé nécessitent avant la phase d'utilisation une étape de labélisation effectuée l'opérateur, qui n'est pas autre chose qu'une part de supervision.

Apprentissage par renforcement : L'apprentissage renforcé est une technique similaire à l'apprentissage supervisé à la différence qu'au lieu de fournir des résultats désirés au réseau (Coulibaly et al ., 1999) on lui accorde plutôt un grade (ou score) qui est une mesure du degré de performance du réseau après quelques itérations, les algorithmes utilisant la procédure d'apprentissage renforcé sont surtout utilisés dans le domaine des systèmes de contrôle (White et Sofge ,1992; Sutton, 1992).

Pour ces trois types d'apprentissage, il y a également un choix traditionnel entre :

L'apprentissage << offline >> : toutes les données sont dans une base d'exemples d'apprentissage qui sont traités simultanément ;

L'apprentissage << on-line >> : Les exemples sont présentés les uns après les autres au fur et à mesure de leur disponibilité.

7.1. La loi de Hebb, un exemple d'apprentissage non supervisé :

La loi de Hebb (1949) s'applique aux connexions entre neurones, comme le représente la figure II.7

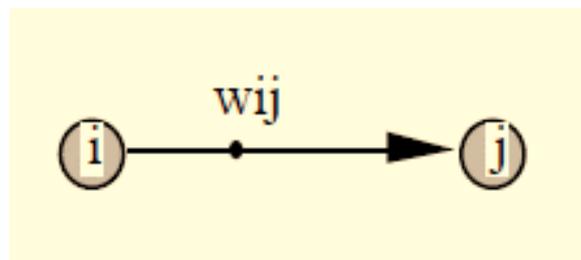


Figure 2.6 : Connexion entre deux neurones

Avec : i le neurone amont ;

j le neurone aval ;

w_{ij} le poids de la connexion.

Elle s'exprime de la façon suivante :

"Si 2 cellules sont activées en même temps alors la force de la connexion augmente". (C'est-à-dire que leurs contacts synaptiques vont être renforcés). On parle de *loi de Hebb* et d'*associationnisme*.

La modification de poids dépend de la coactivation des neurones présynaptiques et post synaptiques, ainsi que le montre la table 1. x_i et x_j sont respectivement les valeurs d'activation des neurones i et j , ∂w_{ij} (dérivée partielle du poids) correspond à la modification de poids réalisée.

Table 2.1. La loi de Hebb

x_i	x_j	∂w_{ij}
0	0	0
0	1	0
1	0	0
1	1	+

La loi de Hebb peut être modélisée par les équations suivantes ($w(t+1)$ est le nouveau poids, $w_{ij}(t)$ l'ancien) :

$$W_{ij}(t+1) = W_{ij}(t) + \partial W_{ij}(t)$$

$$\partial W_{ij}(t) = X_i \cdot X_j$$

(La Coactivité est modélisée comme le produit des deux valeurs d'activation)

L'algorithme d'apprentissage modifie de façon itérative (petit à petit) les poids pour adapter la réponse obtenue à la réponse désirée. Il s'agit en fait de modifier les poids lorsqu'il y a erreur seulement.

1/ Initialisation des poids et du seuil S à des valeurs (petites) choisies au hasard.

2/ Présentation d'une entrée $E_1 = (e_1, \dots, e_n)$ de la base d'apprentissage.

3/ Calcul de la sortie obtenue x pour cette entrée :

$$a = \sum (w_i \cdot e_i) - S \text{ (la valeur de seuil est introduite ici dans le calcul de la somme pondérée)}$$

$$x = \text{signe}(a) \text{ (si } a > 0 \text{ alors } x = +1 \text{ sinon } a \leq 0 \text{ alors } x = -1)$$

4/ Si la sortie x est différente de la sortie désirée d_1 pour cet exemple d'entrée E_1 alors modification des poids (μ est une constante positive, qui spécifie le pas de modification des poids) :

$$w_{ij}(t+1) = w_{ij}(t) + \mu \cdot (x_i \cdot x_j)$$

5/ Tant que tous les exemples de la base d'apprentissage ne sont pas traités correctement (i.e. modification des poids), retour à l'étape 2.

8. Le Perceptron et l'Adaline:

Le Perceptron (Rosenblatt, 1958), qui est la forme la plus simple de réseau de neurones, ce réseau, capable d'apprendre à différencier des formes simples, et à calculer certaines fonctions logiques, est inspiré du système visuel

L'Adaline (ADAPtative LInear NEurone) est une extension du Perceptron (simple), l'idée est de remplacer la fonction seuil sur les neurones de sortie par la fonction identité ($f(x)=x$). L'avantage de cette approche est de ne plus être restreinte à une réponse binaire, et d'offrir tout l'éventail de valeurs possibles.

L'algorithme d'apprentissage utilisé pour le Perceptron est le suivant : en cas d'erreur en sortie le vecteur d'entrée correspondant est ajouté ou retranché au vecteur des poids synaptiques. L'ADALINE, par contre, utilise une règle plus complète car l'erreur est calculée non plus à partir de l'état binaire de la cellule mais à partir de la stimulation que la cellule aurait dû recevoir en entrée (Figure. 7) (Heddam, 2011). De plus cette méthode (dite « Apprentissage de Widrow-Hoff » ou « règle Δ » « Règle de Delta ») permet une réduction du gradient d'erreur à partir de la méthode des moindres carrés. L'erreur quadratique (E) à minimiser s'écrit comme suit :

$$E = (\Psi - \Psi_{\text{désiré}})^2 \text{ (erreur multivaluée dans le cas de l'Adaline)} \quad 1$$

$$E = (Y - Y_{\text{désirée}}) \text{ (erreur binaire dans le cas du perceptron)} \quad 2$$

On note : potentiel désiré $\Psi_{\text{désiré}}$ pour l'Adaline, sortie désirée pour le perceptron $Y_{\text{désirée}}$

La principale limitation de ces deux réseaux provient directement de leur structure : en 1968 Minsky et Papert ont montré, (Minsky et Papert, 1969), que les modèles de réseaux à une couche de connexions et à fonction d'activation linéaire sont incapables de distinguer deux classes si celles-ci ne sont pas linéairement séparables.

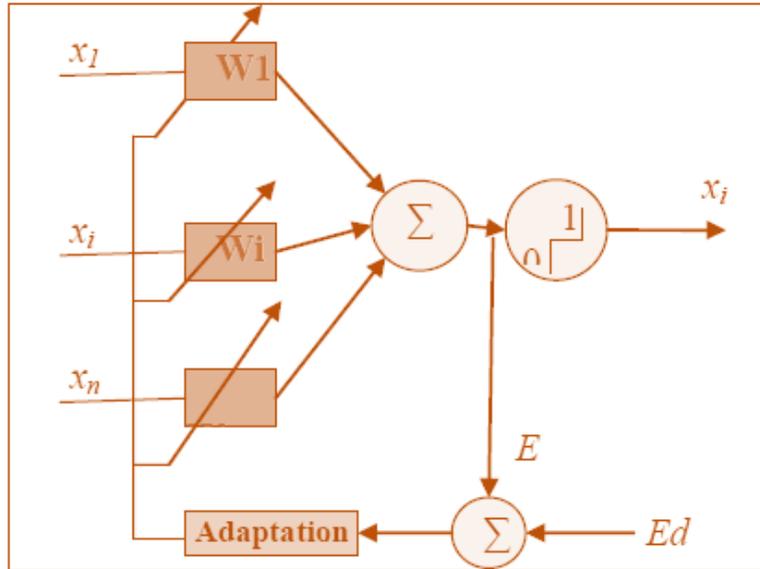


Figure 2.7 : L'ADALINE de Widrow-Hoff

Une solution à cette limitation est les réseaux de neurones considérés comme des approximateurs universels dont les plus étudiés et appliqués en général sont les perceptrons multicouches (Bishop, 1995).

8.1. Le perceptron multicouche:

L'idée principale d'un perceptron multicouches (noté MLP pour Multi Layer Perceptron en anglais) est de grouper des neurones dans une *couche*. En place ensuite bout à bout plusieurs couches et connecte complètement les neurones de deux couches adjacentes. Les entrées des neurones de la deuxième couche sont donc en fait les sorties des neurones de la première couche. Les neurones de la première couche sont reliés au monde extérieur et reçoivent tous le même vecteur d'entrée (c'est en fait l'entrée du réseau). Ils calculent alors leur sorties qui sont transmises aux neurones de la deuxième couche, etc. Les sorties des neurones de la dernière couche forment la sortie du réseau.

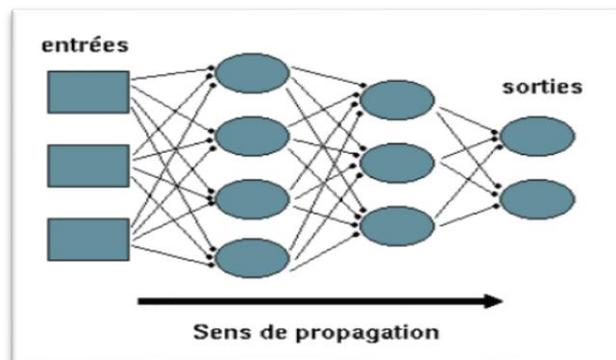


Figure 2.8 : Perceptron multicouches

Donc un neurone d'un perceptron multicouche se caractérise par trois concepts: son état, ses connexions avec d'autres neurones, et sa fonction d'activation.

Algorithmes d'apprentissage supervisé

Pour les réseaux de neurones, les algorithmes les plus utilisés, sont les modèles d'optimisation non linéaires par rapport aux paramètres, les vecteurs des entrées et de la sortie sont centrés :

$$Y = f(X, \theta) + \omega$$

Où f est une fonction de régression (d'activation).

Ces modèles utilisent des méthodes itératives pour obtenir une estimation des paramètres. Leur principe est de se placer en un point, de trouver une direction de descente du coût dans l'espace des paramètres, et de se déplacer d'un pas suivant cette direction, on poursuit cette démarche jusqu'à satisfaction de ce critère d'arrêt (Dreyfus et al., 2004).

9. La Backpropagation :

On sait depuis les années soixante que les possibilités de traitement des réseaux multicouches sont supérieures à celle du Perceptron, cependant l'algorithme d'apprentissage manquait. Pour la couche de sortie, on peut appliquer l'apprentissage du Perceptron, mais comment modifier les poids pour les connexions qui ne sont pas en relation avec un neurone de sortie ?

Le problème est ramené à l'obtention d'une estimation de la valeur désirée pour chaque neurone de la couche cachée. La rétropropagation de gradient est une solution à ce problème.

Cette algorithme très simple et devenu célèbre, est basé sur l'idée suivante : au départ, les poids $\{w_{ij}\}$ sont initialisés à des valeurs aléatoires, le but de la méthode d'apprentissage est de les faire évoluer de façon à ce que le réseau soit capable, étant donné un vecteur d'entrée, de calculer le bon vecteur en sortie.

9.1. Méthode d'optimisation par la Backpropagation :

2.1.1. Algorithme de Newton :

La méthode de Newton utilise la courbure (dérivée seconde) de la fonction de coût pour atteindre le minimum plus rapidement. Elle consiste à minimiser la fonction d'énergie :

$$E(k+1) = E(k) + E'(k)\Delta W(k) + \frac{1}{2}\Delta W(k)^T \ddot{E}(k)\Delta W(k)$$

La modification des paramètres est donnée par :

$$\begin{cases} \theta_k = \theta_{k-1} - H_{k-1}^{-1} \nabla J \\ \text{et } d_{k-1} = -H_{k-1}^{-1} \nabla J \\ u_{k-1} = 1 \text{ (constant)} \end{cases}$$

2.1.2. Algorithme de Levenberg-Maquardt :

La méthode de Levenberg-Maquardt (Levenberg, 1944 ; Maquardt, 1963) est un standard pour l'optimisation de l'erreur quadratique due à ses propriétés de convergence rapide et de robustesse, elle s'appuie sur les techniques des moindres carrés non linéaires et de l'algorithme de Gauss-Newton à voisinage restreint.

10. Conclusion :

Le problème pour un réseau à une ou plusieurs couches cachées, est d'évaluer l'erreur

imputable au poids d'une connexion vers un neurone caché, la solution proposée par plusieurs auteurs (Werbos, 1974 ; Parker, 1985 ; Rumelhart et al., 1986 ; Le Cun ,1987) est de "rétropropager" l'erreur proportionnellement aux poids, la rétro propagation comprend donc deux phases :

- Présentation d'un exemple et propagation des activations vers la sortie
- Calcul de l'erreur en sortie et propagation vers l'ent

Chapitre 3 : La logique floue

1. Introduction :

Au départ théorie, la logique floue s'affirme comme une technique opérationnelle. C'est une extension de la logique classique, Utilisée à côté d'autres techniques de contrôle avancé, elle fait une entrée discrète, mais appréciée dans les automatismes de contrôle industriel.

Les bases théoriques de la logique floue (fuzzy logic) ont été établies au début des années 1965 par le professeur Zadeh de l'université de Californie de Berkeley, en se basant sur sa théorie mathématique des ensembles flous, qui est une généralisation de la théorie des ensembles classiques en introduisant la notion de degré dans la vérification d'une condition, permettant ainsi à une condition d'être dans un autre état que vrai ou faux, la logique floue confère une flexibilité très appréciable aux raisonnements qui l'utilisent, ce qui rend possible la prise en compte des imprécisions et des incertitudes.

Il s'agit d'une approche calquée sur le raisonnement humain, plutôt que sur des calculs rigides pour des problèmes mal définis.

En effet, le mode de raisonnement en logique floue est plus intuitif que la logique classique. Il permet aux concepteurs de mieux appréhender les phénomènes naturels, imprécis et difficilement modélisables.

2. Historique de la logique floue :

Les prémisses de la logique floue, visant à traiter la notion d'incertitude, datent des années 30, il faudra cependant attendre que Zadeh (Zadeh, 1965) introduise le concept de sous-ensembles flous, en 1965, pour assister aux premières grandes avancées dans le domaine. Par la suite, en 1974, Mamdani (Mamdani, 1975) introduisait la commande floue pour la régulation de processus industriels. Initialement, cette théorie a été appliquée dans des domaines non techniques, comme le commerce, les jurys-prudence ou la médecine, dans le but de compléter les systèmes experts, et afin de leur donner l'aptitude de prise de décision. Dès 1975, on trouve les premières applications au niveau des systèmes de réglage. Enfin, dans les années 80, la commande floue connaît un essor considérable au Japon par l'utilisation des produits industriels pour résoudre des problèmes de réglage et de commande, notamment grâce aux travaux de Sugeno pour se répandre ensuite dans le monde entier (Faucher, 2006).

Tout récemment, aussi en Europe, une intense activité de recherche a débuté afin d'introduire le principe du réglage par logique floue.

La logique floue représente une alternative à la logique dite «conventionnelle» introduite par Aristote, où un élément doit absolument appartenir ou non à un ensemble. En contrepartie, la logique floue, permet d'affirmer qu'un élément peut appartenir à un ensemble avec un certain degré d'appartenance (connaissance imparfaites). Il existe différentes théories décrivant la modélisation et le raisonnement à partir des connaissances imparfaites (Ciofolo, 2005), les principales sont les probabilités, les fonctions de croyances (Shafer, 1976), et les sous-ensembles flous (Zadeh, 1965)

associés à la théorie des possibilités (Zadeh, 1965 ; Zadeh, 1973 ; Zadeh, 1978 ; Dubois et Prade, 1980 ; Dubois et Prade, 1988).

3. Rappel sur les ensembles classiques :

La **théorie des ensembles classiques**, malgré son nom compliqué, désigne simplement la branche des mathématiques qui étudie les ensembles. Par exemple, $\{5; 10; 7; 6; 9\}$ est un ensemble d'entiers. $\{0; 1; 2; 3; 4; 5; 6; 7; 8; 9; 10\}$ est l'ensemble des entiers compris entre 0 et 10. $\{'S'; 'd'; 'z'; 'a'\}$ est un ensemble de caractères. $\{"Site"; "du"; "zéro"\}$ est un ensemble de mots. Nous pouvons également créer des ensembles de fonctions, d'hypothèses, de définitions, des ensembles d'individus (c'est-à-dire une population), etc. et même des ensembles d'ensembles.

À noter que dans un ensemble, l'ordre n'a pas d'importance : $\{7; 6; 9\}$ désigne le même ensemble que $\{9; 7; 6\}$. Néanmoins, afin d'améliorer la lisibilité, il est pratique de classer les éléments par ordre croissant, en l'occurrence $\{6; 7; 9\}$. Usuellement, un ensemble est désigné par une lettre en majuscule : ainsi, nous écrirons $A = \{6; 7; 9\}$. L'ensemble vide est noté \emptyset : c'est un ensemble remarquable car il ne contient aucun élément.

Les ensembles sont souvent représentés sous forme graphique, typiquement par des cercles :

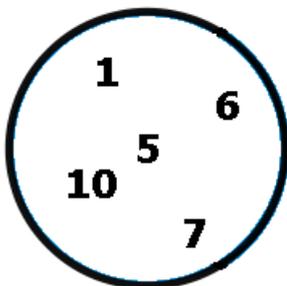


Figure 3.1 : Représentation graphique de l'ensemble $\{1 ; 5 ; 6 ; 7 ; 10\}$

Le concept d'**appartenance** est primordial dans la théorie des ensembles : il désigne le fait qu'un élément fasse partie ou non d'un ensemble. Par exemple, l'entier 7 appartient à l'ensemble $\{6 ; 7 ; 9\}$. À contrario, l'entier 5 n'appartient pas à l'ensemble $\{6; 7; 9\}$. Pour simplifier les choses, l'appartenance est symbolisée par le caractère \in et la non-appartenance par le même symbole, mais barré \notin . Ainsi, nous avons $7 \in \{6 ; 7 ; 9\}$ et $5 \notin \{6 ; 7 ; 9\}$.

Une **fonction d'appartenance** (également appelée fonction indicatrice ou encore fonction caractéristique) est une fonction qui explicite l'appartenance ou non à un ensemble E. Soit f la fonction caractéristique de l'ensemble $E = \{6 ; 7 ; 9\}$, et x un entier quelconque :

$$f(x) = \begin{cases} 1 & \text{si } x \in E \\ 0 & \text{si } x \notin E \end{cases}$$

Cette notion d'appartenance est très importante car la logique floue se base sur le concept d'appartenance floue. Cela signifie simplement que l'on peut appartenir par exemple à 0,8 à un ensemble, contrairement à la théorie des ensembles classiques où comme nous venons de le voir l'appartenance est soit 0 (n'appartient pas) ou 1 (appartient).

Afin de pouvoir manipuler les ensembles classiques, et d'en faire quelque chose d'intéressant, nous définissons un ensemble d'opérations, lesquelles sont très intuitives :

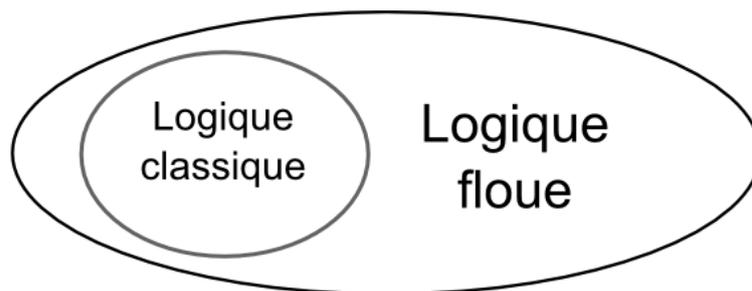
- Réunion de 2 ensembles, notée $A \cup B = \{x \in A \text{ ou } x \in B\}$. $A \cup B$ correspond à la partie en bleu. Par exemple, si $A = \{6; 7; 9\}$ et $B = \{1; 5; 6; 7; 10\}$, alors $A \cup B = \{1; 5; 6; 7; 9; 10\}$.
- Intersection de 2 ensembles, notée $A \cap B = \{x \in A \text{ et } x \in B\}$
Par exemple, si $A = \{6; 7; 9\}$ et $B = \{1; 5; 6; 7; 10\}$, alors $A \cap B = \{6; 7\}$.

4. Les sous-ensembles flous : concepts et définitions

La théorie des sous-ensembles flous permet de s'affranchir du caractère binaire ou booléen de l'appartenance à un ensemble, en introduisant une notion d'appartenance graduelle.

Ainsi, on peut définir un sous-ensemble flou A de S par sa fonction d'appartenance μ_A , qui associe à tout élément x de S une valeur réelle $\mu_A(x)$ dans l'intervalle $[0, 1]$, indiquant le degré d'appartenance de x à A (Ciofolo, 2005).

Pour faire une métaphore en langage ensembliste, la théorie des ensembles classiques n'est qu'un sous-ensemble de la théorie des ensembles flous.



« La théorie des ensembles classiques n'est qu'un sous-ensemble de la théorie des ensembles flous »

Tout simplement un ensemble classique est un ensemble ayant une limite stricte ; prenons un exemple, $A = \{x / u > 5\}$.

Le point $u = 5$ est une limite claire, non ambiguë.

Si u est plus grand que 5, il appartient à A.

Un ensemble flou, est un ensemble sans limite stricte, la notion d'appartenance à un ensemble est graduelle, c'est à dire qu'un élément peut appartenir, plus au moins fortement à cet ensemble.

Ci-après une figure montrant la fonction d'appartenance choisie pour caractériser le sous-ensemble 'bon' de la qualité du service :

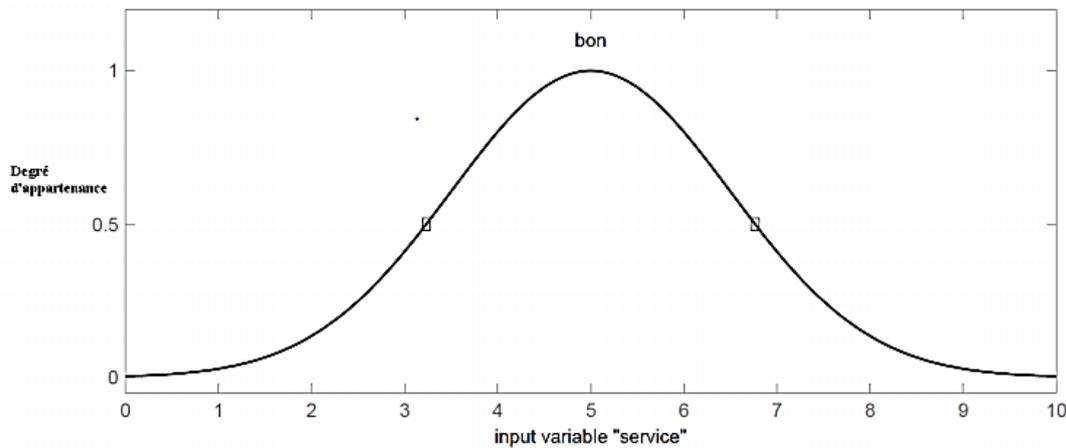


Figure 3.2: Fonction d'appartenance caractérisant le sous-ensemble 'bon' de la qualité du service

DEFINITION : Soit X un ensemble. Un **sous-ensemble flou** A de X est **caractérisé** par une **fonction d'appartenance** $f_A : X \rightarrow [0,1]$. Cette notation veut simplement dire que quel que soit l'entrée X donnée à la fonction f_A , sa sortie est un réel entre 0 et 1. En théorie, il est possible que la sortie soit supérieure à 1, mais en pratique cela n'est quasiment jamais utilisé. Note : cette fonction d'appartenance est l'équivalent de la fonction caractéristique d'un ensemble classique.

La forme de la fonction d'appartenance est choisie arbitrairement en suivant les conseils de l'expert ou en faisant des études statistiques : formes sigmoïde, tangente hyperbolique, exponentielle, gaussienne ou de toute autre nature sont utilisables.

Dans les univers de discours continus, les ensembles flous sont définis analytiquement par des fonctions d'appartenance, afin de faciliter les calculs, elles sont souvent paramétrées (Babuska, 1998).

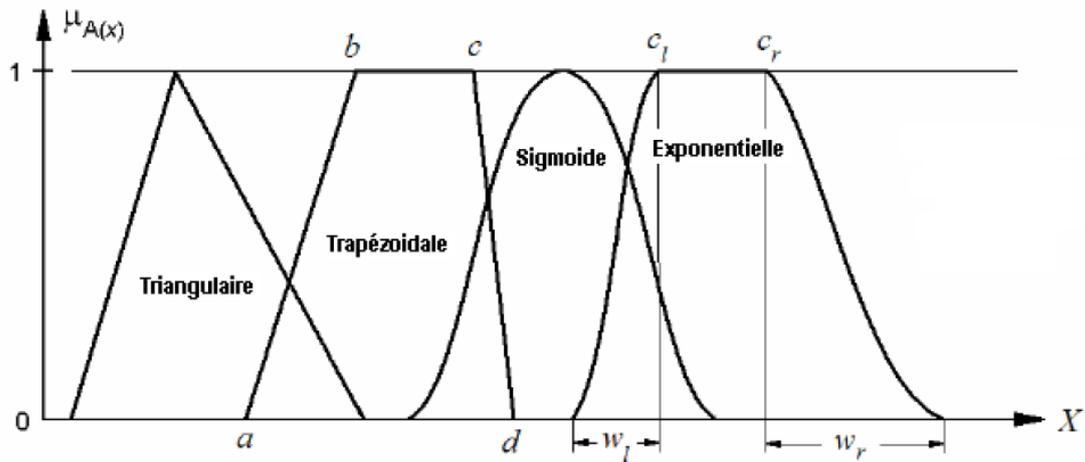


Figure 3 .3: Exemple de fonctions d'appartenance

Fonction d'appartenance trapézoïdale :

$$\mu(x; a, b, c, d) = \max\left(0, \min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right)\right)$$

Où a, b, c, d sont les coordonnées des sommets du trapèze (Figure 3). Il faut remarquer que si $b=c$ on obtiendra une fonction d'appartenance triangulaire.

Fonction d'appartenance exponentielle par morceaux :

$$\mu(x; c_l, c_r, w_l, w_r) = \begin{cases} \exp\left(-\left(\frac{x-c_l}{2w_l}\right)^2\right) & \text{si } x < c_l \\ \exp\left(-\left(\frac{x-c_r}{2w_r}\right)^2\right) & \text{si } x > c_r \\ 1 & \text{les autres cas} \end{cases}$$

Où les paramètres $\{w_l, w_r, c_l \text{ et } c_r\}$ correspondent aux points définis au niveau de la (Figure III.3), pour les conditions $c_l = c_r$ et $w_l = w_r$, on obtiendra la fonction d'appartenance Gaussienne.

5. Éléments caractéristiques des sous-ensembles flous :

Pour pouvoir définir les caractéristiques des ensembles flous, nous redéfinissons et étendons les caractéristiques usuelles des ensembles classiques.

Soit E un ensemble, A un sous-ensemble flou de E et μ_A la fonction d'appartenance le caractérisant.

Bien que la fonction d'appartenance μ_A d'un sous-ensemble flou A permette de le définir entièrement plusieurs éléments caractéristiques sont couramment employés pour les manipuler (Ragot, 2003). Nous les énumérons ici en considérant E comme étant l'univers de référence (Figure III.4).

DEFINITION 1 : La **hauteur** de A , notée $h(A)$, correspond à la borne supérieure de l'ensemble d'arrivée de sa fonction d'appartenance : $h(A) = \sup \{ \mu_A(x) | x \in A \}$.

DEFINITION 2 : A est dit **normalisé** si et seulement si $h(A) = 1$. En pratique, il est extrêmement rare de travailler sur des ensembles flous non normalisés.

DEFINITION 3 : Le **support** de A est l'ensemble des éléments de E appartenant au moins un peu à A . Autrement dit, c'est l'ensemble $\text{supp}(A) = \{ x \in E | \mu_A(x) > 0 \}$.

DEFINITION 4 : Le **noyau** de A est l'ensemble des éléments de E appartenant totalement à A . Autrement dit, c'est l'ensemble $\text{noy}(A) = \{ x \in E | \mu_A(x) = 1 \}$. Par construction,

$$\text{noy}(A) \subseteq \text{supp}(A).$$

DEFINITION 5 : Une **α -coupe** de A est le sous-ensemble classique des éléments ayant un degré d'appartenance supérieur ou égal à α : $A_\alpha = \{ x \in E | \mu_A(x) \geq \alpha \}$.

DEFINITION 6 : La **cardinalité** $|A|$ d'un sous-ensemble flou A est la quantité (floue) d'éléments de E qui appartiennent à A : $|A| = \sum_{x \in E} \mu_A(x)$.

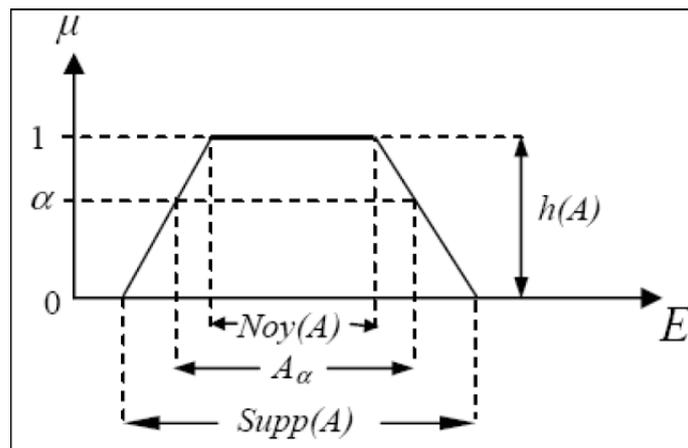


Figure 3.4: Principaux éléments caractéristiques d'un sous-ensemble flou A

6. Principe fondamentale de la logique floue :

Comparativement à la logique classique, les bases théoriques de la logique floue sont établies de manière à pouvoir traiter des variables inexacts de valeurs comprises entre 0 et 1, par contre la logique de Boole dont les variables ne peuvent prendre que les valeurs 0 et 1.

À titre d'exemple la classification des personnes à travers leur âge par les deux logiques présenté dans la **figure 3. 5 graphe (1)**.

Fait apparaître que :

1. La logique classique (logique de Boole) n'admet pour les variables que les valeurs 0 et 1, qui font que les personnes âgées de moins de 30 ans sont systématiquement jeunes et les plus de 50 ans sont âgées, sans pour autant que, cette classification soit logique.
2. Alors que la logique floue, dont les variables peuvent prendre n'importe quelle valeur comprise entre 0 et 1, permet de tenir compte du passage progressif de l'individu d'un âge à un autre, on parle alors, de fonction d'appartenance μ .

Les limites ne varient pas soudainement, mais progressivement, la figure 5 montre une classification possible ; une personne de 25 ans appartient à l'ensemble (jeune) avec une valeur $\mu = 0.75$ de la fonction d'appartenance, et à l'ensemble (entre deux âges) avec $\mu = 0.25$. Par contre une personne de 70 ans appartient avec une valeur $\mu = 1$ de la fonction d'appartenance à l'ensemble (âgé).

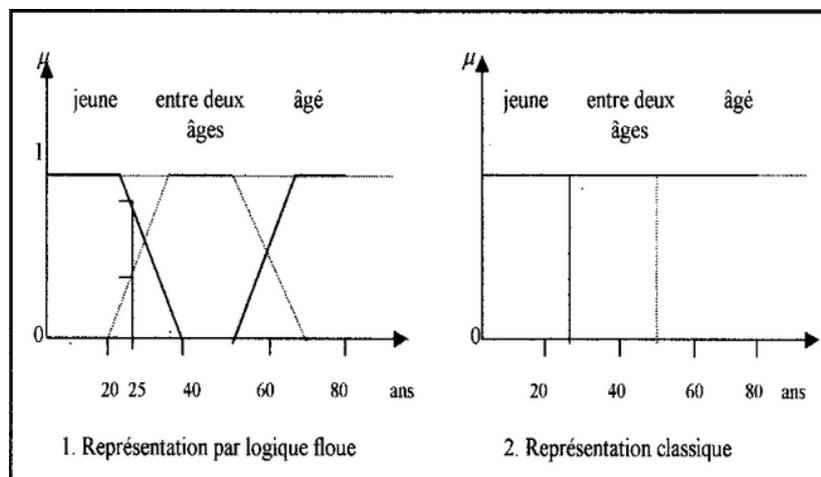


Figure 3. 5: Deux représentations possibles classique et par logique floue des personnes selon leur âge

7. Les variables linguistiques :

Le concept de fonction d'appartenance vu précédemment nous permettra de définir des systèmes flous en langage naturel, la fonction d'appartenance faisant le lien entre logique floue et variable linguistique que nous allons définir à présent.

DEFINITION : Soit V une variable (qualité du service, montant du pourboire, etc.), X la plage de valeurs de la variable (par exemple, entre 0 et 30€ pour le pourboire) et T_V un ensemble fini ou infini de sous-ensembles flous. Une **variable linguistique** correspond au triplet (V, X, T_V) .

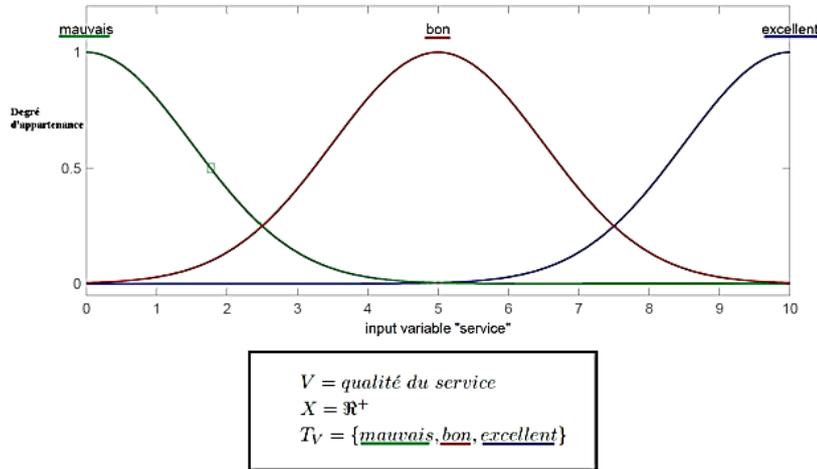


Figure 3.6 : Exemple de variable linguistique ‘qualité de service

8. Les opérateurs flous :

Les opérateurs disponibles dans le cadre de la théorie des sous-ensembles flous sont nombreux, et ont une propriété importante du point de vue de la fusion d'informations, ils donnent un résultat qui a la même interprétation en termes d'imprécision que leurs entrées, ce qui signifie que la décision peut être reportée à la fin du processus de combinaison, évitant ainsi des décisions partielles qui pourraient conduire à des imperfections (Ciofolo, 2005).

Contrairement aux définitions des propriétés des ensembles classiques qui sont toujours les mêmes, la définition des opérateurs sur les ensembles flous est choisie, à l'instar des fonctions d'appartenance. Voici les deux ensembles d'opérateurs pour le complément (NON), l'intersection (ET) et l'union (OU) utilisés le plus couramment :

Table 3.1 : Les opérateurs de la logique floue

Dénomination	Intersection ET : $\mu_{A \cap B}(x)$	Réunion OU : $\mu_{A \cup B}(x)$	Complément NON : $\mu_{A^c}(x)$
Opérateurs de Zadeh MIN/MAX	Min $(\mu_A(x), \mu_B(x))$	Max $(\mu_A(x), \mu_B(x))$	$1 - \mu_A(x)$
Probabiliste PROD/PROBOR	$\mu_A(x) \times \mu_B(x)$	$\mu_A(x) + \mu_B(x) - \mu_A(x) \times \mu_B(x)$	$1 - \mu_A(x)$

Avec les définitions usuelles des opérateurs flous, nous retrouvons toujours les propriétés de commutativité, distributivité et associativité des opérateurs classiques. Cependant, nous relevons deux exceptions notables :

- en logique floue, le principe du tiers exclu est contredit : $A \cup A^{\bar{}} \neq X$, autrement dit $\mu_{A \cup A^{\bar{}}}(x) \neq 1$.
- en logique floue, un élément peut appartenir à A et non A en même temps : $A \cap A^{\bar{}} \neq \emptyset$, autrement dit $\mu_{A \cap A^{\bar{}}}(x) \neq 0$. Notons que ces éléments correspondent à l'ensemble $\text{supp}(A) - \text{noy}(A)$.

9. Le raisonnement en logique floue :

En logique classique, les raisonnements sont de la forme :

$$\begin{cases} \text{Si } p \text{ alors } q \\ p \text{ vrai alors } q \text{ vrai} \end{cases}$$

En logique floue, le raisonnement flou, également appelé raisonnement approximatif, se base sur des **règles floues** qui sont exprimées en langage naturel en utilisant les variables linguistiques dont nous avons donné la définition précédemment. En logique, une règle est sous la forme : Si [prémisses] alors [conclusion].

En logique floue, les règles auront des formes telles que :

Si $x \in A$ et $y \in B$ alors $z \in C$

Si $x \in A$ alors $z \in C$

Si $x \in A$ ou $y \in B$ alors $z \in C$

Avec A, B et C des ensembles flous.

EXEMPLE : *Si (la qualité de la nourriture est délicieuse), alors (le pourboire sera élevé)*'. (ce qui correspond à la forme *Si $x \in A$ alors $z \in C$*).

10. Conclusion :

Au cours des définitions, nous avons vu que le concepteur d'un système flou doit faire un nombre de choix important. Ces choix se basent essentiellement sur les conseils de l'expert ou sur l'analyse statistique des données passées, en particulier pour définir les fonctions d'appartenance et la matrice des décisions.

Voici un aperçu synoptique d'un système flou :

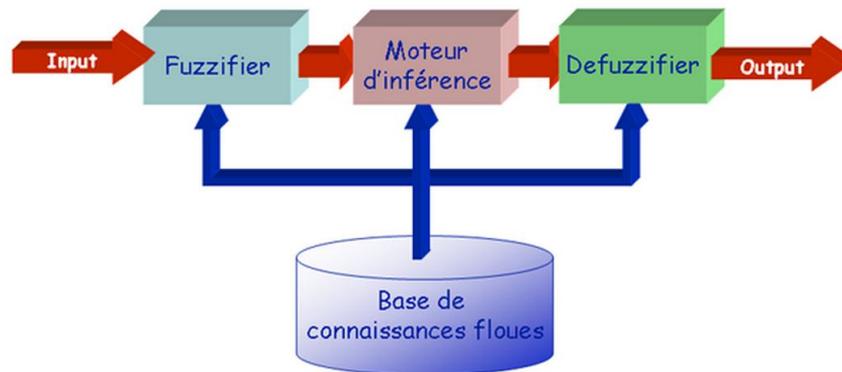


Figure 3.7: Composantes d'un système flou

Un système d'inférence flou est essentiellement composé de deux blocs fonctionnels différents (Renaud, 1999) :

1. Une base de connaissance constituée d'une base de règles floues et d'une base de données définissant les fonctions d'appartenance des ensembles flous ;
2. Un mécanisme d'inférence flou qui détermine la sortie du système sous la forme d'un ensemble flou ;

On peut y ajouter deux blocs (Figure .7) :

3. Une interface de fuzzification, qui transforme des entrées ponctuelles en ensembles flous, le cas échéant ;
4. Une interface de défuzzification, qui transforme la sortie floue en sortie ponctuelle.

Les deux premiers blocs définissent des relations entre les entrées et sorties floues du système. Les entrées du système $\{x_1, \dots, x_r\}$ peuvent être fixes ou floues.

11. La fuzzification :

La fuzzification est la première étape du traitement flou, elle consiste à caractériser au moyen de sous-ensembles flous les valeurs des variables d'entrées représentant l'état du système. La fuzzification n'est qu'une étape technique permettant d'introduire des entrées observées fixes dans le mécanisme de raisonnement proprement dit en convertissant artificiellement ces entrées en ensembles flous (*fuzzification symbolique*), si ces données sont déjà définies par des ensembles flous, cette étape est donc inutile.

Dans le cas contraire (*fuzzification numérique*), on distingue la fuzzification dite «singleton» qui consiste simplement à transformer un élément fixe x_k de X_k en un sous-ensemble classique $\{x_k\}$ de X_k , notée $x_{k\sim}$. Ce type de fuzzification ne fait pas intervenir l'imprécision éventuelle des données, la méthode du singleton demeure la plus simple, et la plus fréquemment employée. Une autre approche, la fuzzification «non singleton» intègre l'imprécision ou le bruit dû à l'observation,

en représentant l'entrée x_k sous la forme d'un ensemble flou quelconque \tilde{x}_k , comme par exemple un nombre flou gaussien centré en x_k

(Renaud, 1999). Dans la littérature des SIF, la fuzzification singleton est de loin la plus utilisée, quelques rares cas illustrent cependant une fuzzification non-singleton (Palm, 1994 ; Galichet, 1994).

12. La défuzzification :

La défuzzification transforme la partie floue issue de l'inférence en une valeur numérique $y \in Y$, le résultat de l'inférence peut être soit une partie floue numérique, soit une partie floue symbolique, afin de les transformer en une valeur numérique, on aura besoin de deux types de défuzzification : une défuzzification numérique δ_1 , et une défuzzification symbolique δ_2 (Nakoula, 1997), le but de la défuzzification est de trouver une sortie précise $y^* = y_0$ qui corresponde au mieux à la caractérisation floue.

Comme pour tous les opérateurs flous, le concepteur du système flou doit choisir parmi plusieurs définitions possibles de défuzzification. Une liste détaillée peut être consultée dans

[Leekwijck and Kerre, 1999]. Nous allons présenter brièvement les deux principales méthodes de défuzzification : la méthode moyenne des maxima (MM) et la méthode du centre de gravité (COG). Cependant la méthode la plus courante et la plus répandue est celle du centre de gravité.

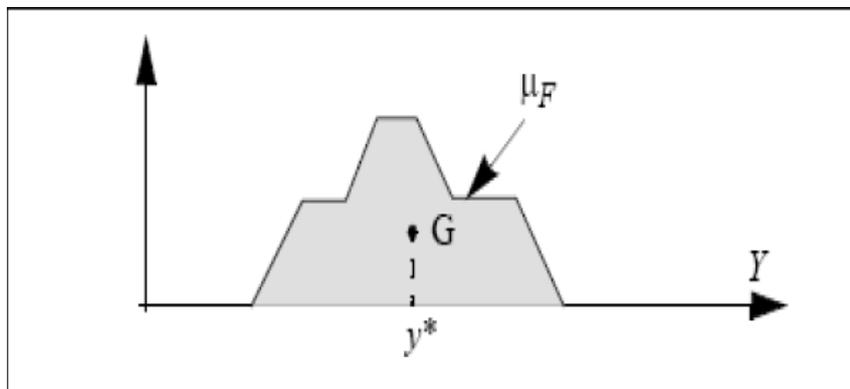


Figure 3.8: Défuzzification par le centre de gravité

Soit $F \in F(Y)$, la partie floue numérique issue de l'inférence. La valeur numérique y^* produite par l'application de la méthode du centre de gravité est donnée par :

$$y^* = \delta_1(F) = \frac{\int_{y \in Y} \mu_F(y) \cdot y \cdot dy}{\int_{y \in Y} \mu_F(y) \cdot dy} \quad \text{si } Y \text{ est continu}$$

$$y^* = \delta_1(F) = \frac{\sum_{y \in Y} \mu_F(y) \cdot y}{\sum_{y \in Y} \mu_F(y)} \quad \text{si } Y \text{ est non (discret)}$$

La valeur y^* produite par la défuzzification δ_1 est définie comme étant la coordonnée y du centre de gravité (G) de la surface située sous μ_F (Figure).

13. Association d'un modèle flou neuronal : l'approche neurofloue

Les systèmes neuro-flous permettent de combiner les avantages de deux techniques complémentaires. Les systèmes flous fournissent une bonne représentation des connaissances. L'intégration de réseaux de neurones au sein de ces systèmes amélioré leurs performances grâce à la capacité d'apprentissage des réseaux de neurones. Inversement, l'injection de règles floues dans les réseaux de neurones, souvent critiquées pour leur manque de lisibilité, clarifie la signification des paramètres du réseau et facilite leur initialisation, ce qui représente un gain de temps de calcul considérable pour leur identification (Renaud, 1999).

On peut représenter cette analyse sous la forme suivante :

Les neurones de la première couche cachée vont représenter les différentes zones de la logique floue. Les différentes sorties de cette couche donneront donc les degrés d'appartenance correspondant aux différentes zones. On introduit alors dans une deuxième couche cachée les règles d'inférence de la logique floue.

Ce qui nous permet d'obtenir la règle: **si x ET si y ALORS Conséquence**.

La couche de sortie va nous donner le résultat qui peut être obtenu par défuzzification. Le schéma ci-dessus met en évidence le parallélisme massif entre le raisonnement flou et la représentation connexionniste. On retrouve les réseaux de neurones lors de l'incorporation des poids par apprentissage.



On peut ajouter des poids plus ou moins prédéfinis ce qui nous donnera un basculement entre la connaissance explicite (Logique Floue : Les poids sont tous prédéfinis) et le raisonnement implicite (Réseaux de Neurones : Poids établis par apprentissage).

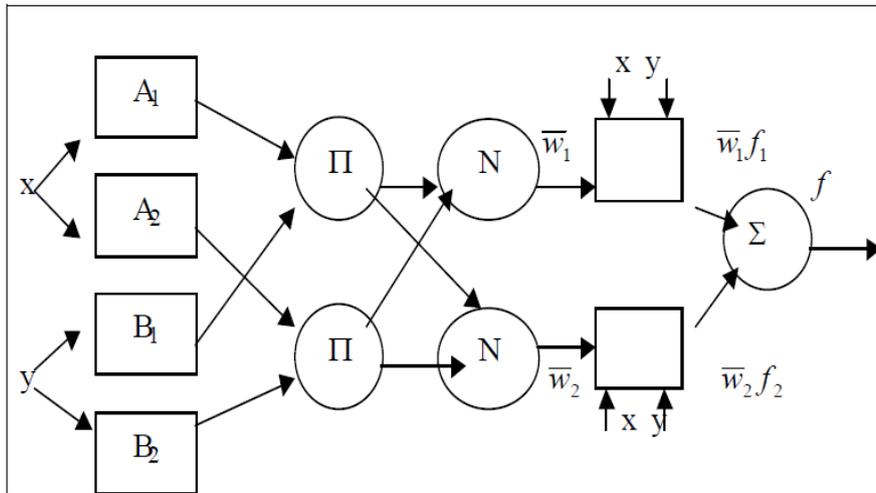


Figure 3.9 : Modèle adaptatif ANFIS

Conclusion :

Cette association semble apporter au niveau de l'apprentissage qui est ainsi moins complexe (certains poids étant préétablis), et au niveau de la compréhension du cheminement de la connaissance dans les réseaux de neurones.

***Chapitre 4 : Élaborations des
modèles de prévisions***

1. Introduction

Les prévisions hydrologiques ont pour objectif de permettre une planification plus éclairée des interventions, tant pour des situations de crue ou d'étiage que dans des conditions hydrologiques plus courantes.

Pour le cas du barrage de la Cheffia on a opté sur la prévision des apports, des lames d'eau évaporées et des volumes tributaires des entrées et des sorties;

Le choix de ces deux variables se justifie, par l'intérêt apporté par leur prévision dans le domaine de gestion des ressources en eau. Par ailleurs, la distinction de ces deux variables par leur type, nous ramène à confronter deux problèmes de prévision différents. Étant cyclique, les lames évaporées est facilement prévue. À l'opposé, les apports présentent un phénomène complexe, et très difficile à prévoir, particulièrement pour une région à climat semi-aride connue par ses saisons de pluie d'une extrême irrégularité et susceptible d'interruptions qui peuvent durer plusieurs semaines.

2. L'architecture

L'unité de base de chaque réseau de neurone est le neurone formel. Il est constitué d'un intégrateur qui effectue une somme pondérée de ses entrées.

Le perceptron multicouches est un réseau constitué de plusieurs couches disposées l'une après l'autre. Chaque couche contient plusieurs neurones.

Dans ce cas-là, le fonctionnement de chaque neurone est le même. Une somme pondérée doit être effectuée, avant de passer au calcul de la sortie a_j du neurone j par la fonction de transfert f . On peut résumer le calcul de la sortie du a_j par l'expression ci-dessous :

$$a_j = f(\sum w_{ij} \times p_i + b_j)$$

Où :

a_j est la sortie du neurone j ;

f est sa fonction d'activation ;

p_i est un élément du vecteur d'entrée ;

w_{ij} est la connexion (poids) reliant le neurone j à l'entrée p_i ;

b_j est le biais.

Ainsi, pour chaque couche k constituée des s neurones, les vecteurs de sortie (a_k) et de biais (b_k) du même taille s . Les poids w_{ij} forment à leur tour une matrice de dimension $s.r$.

Le vecteur d'entrée peut être le stimuli si c'est le cas de la première couche, comme il peut être le vecteur de sortie de la couche précédente s'il s'agit d'une couche intermédiaire.

La fonction d'activation adoptée dans ce type de réseaux de neurones, est souvent la fonction sigmoïde.

3. Le perceptron multicouches :

Dans le cadre de cette étude, un perceptron multicouches ayant 2, 5, et 10 neurones dans sa couche cachée est adopté comme modèle de prévision. Ce modèle est détaillé par la suite.

4. L'apprentissage :

L'apprentissage des perceptrons multicouches est un apprentissage supervisé par l'algorithme de rétro propagation de l'erreur (Backpropagation).

Il consiste en la modification des poids des connexions entre les différentes couches, par rapport à une erreur calculée en sortie de réseau: on dit alors que l'erreur est rétro propagée (Davallo, 1991).

5. Programmation du modèle neuronale:

Afin d'élaborer le modèle des réseaux de neurones, nous avons opté sur le perceptron multicouches.

La programmation de ce modèle a été faite à l'aide du logiciel MATLAB version R2011a.

Nos essais consistaient à élaborer plusieurs modèles de réseaux de neurones, ceci en modifiant le nombre de couches et le nombre de neurones dans chaque couche.

Le nombre de règles flous a été pris égale à 2, deux règles floues ont été suffisante pour obtenir les résultats qui sont à votre disposition. Si on passe à trois règles floues, le temps de calcul augmente exponentiellement, et le modèle de prévision perd de son attrait.

5.1. Critères de performance :

Pour accepter ou rejeter la prévision, on s'est basé sur : la racine carrée de l'erreur quadratique entre valeur observée et valeur estimée (RMSE) et le coefficient de détermination (r^2).

5.2. Résultats et interprétations :

Une prévision robuste, c'est-à-dire qui prend en compte toutes les corrélations disponibles et les interdépendances existant dans les séquences temporelles, est une tâche difficile. (SOUAG, 2007).

C'est pour cela, on a procédé d'une prévision avec et sans correction.

Dont les paramètres qui changent sont :

1/ Le type de variable cyclique (lames évaporées) ou aléatoire (apports) ou tributaire des entrées et des sorties (volumes) :

Tel que :

A : désigne apport

L : désigne lames évaporée

V : désigne volume

2/ La taille du fichier de départ.

3/ Pourcentage calage : on a pris 30%, 60% et 80% calage.

4/ Le nombre de décalage optimum (D):

On a testé :

Deux (2) décalages : prévoir la variable en temps $t+1$ en connaissant la variable en t ;

Quatre (4) décalages: prévoir la variable en temps $t+1$ en connaissant la variable en t , $t-1$ et $t-2$.

5/ la technique de prévision : Réseaux de neurones ou logique floue.

6/ Nombre de couches cachées (1 ou 2) pour les réseaux de neurones (N).

7/ Le choix du nombre de neurones dans chaque couche : 2, 5, 10, et 20 neurones.

8/Le nombre de règle floue (RF): deux (2) règles floues.

9/Le pas de temps :

M : désignant le pas de temps mensuel ;

J : désignant le pas de temps journalier.

Exemple : AM2N1D30 : prévision des apports mensuels par 2 neurones, avec un (1) décalage, en prenant 30% du fichier calage

6. Prévisions mensuelles par les réseaux de neurones :

6.1. Prévision mensuelle des volumes:

Un (1) décalage :

Période calage :

Table 4.1: Résultats de la prévision mensuelle des volumes avec un décalage (calage)

variante	VM2N2D30	VM2N2D60	VM2N2D80	VM5N2D30	VM5N2D60	VM5N2D80
RMSE	1.4086	1.4072	1.4360	13.8223	14.7366	14.2979
r^2	0.9988	0.9988	0.9988	0.8679	0.8676	0.8776

VM10N2D30	VM10N2D60	VM10N2D80
14.3807	14.4956	14.3164
0.8574	0.8729	0.7866

D'après les résultats présentés dans le tableau on remarque que les résultats obtenus avec deux neurones sont mieux que ceux obtenus avec 5 et 10 neurones, tel que, lorsqu'on a pris 30 %, 60%

et 80% avec 2 neurones on trouve que les résultats sont relativement similaires ainsi que le meilleur modèle de prévision en phase de calage est celui avec de neurones et 60% de fichier calage.

Période validation :

Table 4.2: Résultats de la prévision mensuelle des volumes avec un décalage (validation)

variante	VM2N2D30	VM2N2D60	VM2N2D80	VM5N2D30	VM5N2D60	VM5N2D80
RMSE	1.3250	1.2581	0.9005	17.1494	13.4699	12.0839
R ²	0.9987	0.9986	0.9988	0.7994	0.8366	0.7861

VM10N2D30	VM10N2D60	VM10N2D80
17.1092	12.9720	12.1804
0.8163	0.8490	0.8769

En phase de validation le meilleur modèle est celui avec deux neurones et 80 % du fichier calage.

Trois décalage (3) décalages :

Période de calage :

Table 4.3 : Résultats de la prévision mensuelle des volumes avec trois décalages (calage)

variant	VM2N4D3	VM2N4D6	VM2N4D8	VM5N4D3	VM5N4D6	VM5N4D80
e	0	0	0	0	0	
RMSE	1.2279	1.2722	1.3216	9.7746	10.5769	10.9471
r ²	0.9990	0.9990	0.9990	0.9353	0.9328	0.9288

VM10N4D30	VM10N4D60	VM10N4D80
9.9643	10.3552	10.6846
0.9343	0.9348	0.9326

Nous remarquons que le modèle avec deux neurones donne de bons résultats, une fois on augmente le nombre de neurones à 5 et 10 neurones, on voit que le modèle est incapable de prévoir les volumes mensuels.

Période de validation :

Table 4.4: Résultats de la prévision mensuelle des volumes avec un décalage (validation)

variante	VM2N4D30	VM2N4D60	VM2N4D80	VM5N4D30	VM5N4D60	VM5N4D80
RMSE	1.4418	1.1765	0.9990	11.3834	9.5389	8.8736
R ²	0.9985	0.9988	0.8142	0.9136	0.9172	0.8820

VM10N4D30	VM10N4D60	VM10N4D80
12.9525	9.7735	9.9004
0.8900	0.9134	0.8743

On tire la même conclusion qu'avant pour la prévision avec trois décalages soit en phase de calage ou de validation.

6.2. Prévision mensuelle des lames évaporées

Un (1) décalage :

Période calage :

Table 4.7 : Résultats de la prévision mensuelle des lames évaporées avec un décalage (calage)

variante	LM2N2D30	LM2N2D60	LM2N2D80	LM5N2D30	LM5N2D60	LM5N2D80
RMSE	36.5958	34.3658	57.4781	36.5752	35.2289	33.1476
r ²	0.6640	0.6607	0.0876	0.6641	0.6493	0.6647

LM10N2D30	LM10N2D60	LM10N2D80
39.4226	34.1034	33.3678
0.6375	0.6644	0.6599

Nous remarquons que la prévision des lames évaporées mensuelles avec les réseaux de neurones ne donne pas de bons résultats quelque soit la variante choisie.

Période validation :

Table 4.8 : Résultats de la prévision mensuelle des lames évaporées avec un décalage (validation)

variante	LM2N2D30	LM2N2D60	LM2N2D80	LM5N2D30	LM5N2D60	LM5N2D80
RMSE	30.9061	29.2916	46.4649	30.6408	29.0205	28.0001
r ²	0.6458	0.6506	0.2896	0.6552	0.6549	0.6382

LM10N2D30	LM10N2D60	LM10N2D80
33.3773	29.2551	28.6520
0.6321	0.6525	0.6273

La même conclusion tirée pour la phase calage est constaté dans la phase validation avec un (1) décalage.

Trois (3) décalages :

Période de calage :

Table 4.9 : Résultats de la prévision des lames évaporées mensuelles avec trois décalages (calage)

Variante	LM2N4D30	LM2N4D60	LM2N4D80	LM5N4D30	LM5N4D60	LM5N4D80
RMSE	27.3442	23.6698	11.2580	23.2672	23.3562	20.2372
r ²	0.8180	0.8393	0.9352	0.8682	0.8441	0.8755

LM10N4D30	LM10N4D60	LM10N4D80
29.5180	19.6627	17.5135
0.8350	0.8908	0.9074

Période de validation :

Table 4.10: Résultats de la prévision mensuelle des lames évaporées avec trois décalages (validation)

variante	LM2N4D30	LM2N4D60	LM2N4D80	LM5N4D30	LM5N4D60	LM5N4D80
RMSE	25.5029	27.34420	10.8300	23.2672	19.6566	20.1511
r ²	0.8190	0.8434	0.8929	0.8339	0.8516	0.8242

LM10N4D30	LM10N4D60	LM10N4D80
27.5117	21.2954	21.6632
0.7424	0.8343	0.8074

6.3. Prévisions mensuelles des apports

Un (1) décalage :

Période calage :

Table 4.11 : Résultats de la prévision mensuelle des apports avec un décalage (calage)

variante	AM2N2D30	AM2N2D60	AM2N2D80	AM5N2D30	AM5N2D60	AM5N2D80
RMSE	14.6614	12.7151	13.5460	0.1871	13.6662	13.0372
r ²	0.1116	0.2775	0.2662	13.4624	0.2667	0.3148

AM10N2D30	AM10N2D60	AM10N2D80
12.4758	12.4800	13.2320
0.3703	0.3086	0.2960

Période validation :

Table 4.12 : Résultats de la prévision mensuelle des apports avec un décalage (validation)

variante	AM2N2D30	AM2N2D60	AM2N2D80	AM5N2D30	AM5N2D60	AM5N2D80
RMSE	17.7532	18.2218	21.2639	16.7853	19.1087	20.1269
r ²	0.0826	0.3122	0.2264	0.1871	0.2858	0.3012

AM10N2D30	AM10N2D60	AM10N2D80
17.3488	17.4967	20.2113
0.1912	0.3316	0.3116

Trois (3) décalages :

Période de calage :

Table 4.13 : Résultats de la prévision mensuelle des apports avec trois décalages (calage)

variante	AM2N4D30	AM2N4D60	AM2N4D80	AM5N4D30	AM5N4D60	AM5N4D80
RMSE	12.2374	13.6333	13.1511	11.5470	12.2389	12.5325
r ²	0.4284	0.1649	0.3072	0.4456	0.3309	0.3699

AM10N4D30	AM10N4D60	AM10N4D80
11.7365	13.4629	12.9830
0.4350	0.3396	0.3318

Période de validation :

Table 4.14: Résultats de la prévision mensuelle des apports avec un décalage (validation)

variante	AM2N4D30	AM2N4D60	AM2N4D80	AM5N4D30	AM5N4D60	AM5N4D80
RMSE	17.3437	19.9119	19.8438	16.3785	19.6743	20.1882
r ²	0.2271	0.1219	0.3170	0.2386	0.1634	0.2884

AM10N4D30	AM10N4D60	AM10N4D80
18.4900	19.4494	20.1616
0.1390	0.2698	0.3346

Conclusion :

Dans tous ce qui précède nous remarquons que résultats de prévision par les réseaux de neurones des volumes et des lames évaporées sont très bon car on trouve le coefficient de détermination supérieur à 80 %, alors que la prévision des apports était une tâche très difficile, voire impossible a donné de mauvais résultats.

7. Prévisions mensuelles par la logique floue

7.1. Prévision mensuelle des lames évaporées

Un (1) décalage:

Période calage :

Table 4.15 : Résultats de prévisions par la logique floue des lames évaporées avec un décalage (calage)

variante	LM2RF2D30	LM2RF2D60	LM2RF2D80
RMSE	36.2044	34.1315	33.0778
r ²	0.6691	0.6633	0.6657

Période validation :

Table 4.16 : Résultats de prévision par la logique floue des lames évaporées avec un décalage (validation)

variante	LM2RF2D30	LM2RF2D60	LM2RF2D80
RMSE	30.3497	28.8585	27.9872
r^2	0.6556	0.6582	0.6363

Trois (3) décalages :

Période calage :

Table 4.17: Résultats de prévision par la logique floue des Lames évaporées avec trois décalages (calage)

Variante	LM2RF4D30	LM2RF4D60	LM2RF4D80
RMSE	17.8886	18.2182	17.8437
r^2	0.9218	0.9045	0.9031

Période validation :

Table 4.18 : Résultats de prévision par la logique floue des lames évaporées avec trois décalages (validation)

variante	LM2RF4D30	LM2RF4D60	LM2RF4D80
RMSE	23.5747	22.7360	26.0236
r^2	0.8202	0.8122	0.7487

7.2. Prévision mensuelle des volumes:

Un (1) décalage :

Période calage :

Table 4.19 : Résultats de prévision par la logique floue des volumes avec un décalage (calage)

variante	V2RF2D30	V2RF2D60	V2RF2D80
RMSE	13.6912	14.3459	14.1066
r^2	0.8704	0.8741	0.8798

Période validation :

Table 4.20 : Résultats de prévision par la logique floue des volumes avec un (1) décalage (validation)

variante	V2RF2D30	V2RF2D60	V2RF2D80
RMSE	14.0073	12.6252	11.7110
r^2	0.8594	0.8572	0.7924

Trois (3) décalages :

Période calage :

Table 4.21 : Résultats de prévision par la logique floue des volumes avec trois décalages (calage)

variante	V2RF4D30	V2RF4D60	V2RF4D80
RMSE	7.3901	9.3587	9.2914
r^2	0.7064	0.9467	0.9481

Période validation :

Table 4.22: Résultats de prévision par la logique floue des volumes avec trois décalages (validation)

variante	V2RF4D30	V2RF4D60	V2RF4D80
RMSE	20.7140	9.3232	8.0765
r^2	0.9629	0.9225	0.9054

7.3. Prévisions mensuelles des apports

Un (1) décalage :

Période calage :

Table 4.23 : Résultats de prévision des apports par la logique floue avec un décalage (calage)

variante	AM2RF2D30	AM2RF2D60	AM2RF2D80
RMSE	12.5873	12.5413	13.2205
r^2	0.3450	0.2855	0.2953

Période validation :

Table 4.24 : Résultats de prévision des apports par la logique floue avec un décalage (validation)

variante	AM2RF2D30	AM2RF2D60	AM2RF2D80
RMSE	15.9646	19.4507	21.8753
r^2	0.2610	0.2282	0.2404

Trois (3) décalages :

Période calage :

Table 4.25 : Résultats de prévision des apports par la logique floue avec trois décalages (calage)

variante	AM2RF4D30	AM2RF4D60	AM2RF4D80
RMSE	9.1149	10.7976	11.9521
r^2	0.6543	0.4710	0.4249

Période validation :

Table 4.26 : Résultats de prévision des apports par la logique floue avec trois décalages (calage)

variante	AM2RF4D30	AM2RF4D60	AM2RF4D80
RMSE	968.9206	494.2516	150.8098
r^2	0.0299	0.0014	0.0000

Conclusion :

Nous remarquons que les résultats trouvés pour la prévision des volumes par les réseaux de neurones sont mieux que ceux trouvés par la par la logique floue. Quoiqu'en utilisant 80 % du fichier calage avec trois (3) décalage on trouve de bon résultat. Et c'est la même conclusion à tirer qu'avant pour les trois variables.

8. Les prévisions journalières par les réseaux de neurones :

8.1. Prévisions journalières des apports :

Un (1) décalage :

Période calage :

Table 4.27 : Résultats de prévision des apports journaliers avec un décalage (calage)

variante	AJ2N2D30	AJ2N2D60	AJ2N2D80	AJ5N2D30	AJ5N2D60	AJ5N2D80
RMSE	0.7130	0.7764	0.8198	0.7165	0.7765	0.8189
r^2	0.4015	0.3702	0.3714	0.3956	0.3704	0.3728

AJ10N2D30	AJ10N2D60	AJ10N2D80
0.7756	0.7762	0.8181
0.2998	0.3704	0.3739

Période validation :

Table 4.28 : Résultats de prévision des apports journaliers avec un décalage (validation)

variante	AJ2N2D30	AJ2N2D60	AJ2N2D80	AJ5N2D30	AJ5N2D60	AJ5N2D80
RMSE	1.0229	1.1391	1.3178	1.0385	1.1472	1.3036
r^2	0.3456	0.3489	0.3346	0.3236	0.3360	0.3547

AJ10N2D30	AJ10N2D60	AJ10N2D80
1.1215	1.1395	1.3726
0.2341	0.3457	0.2610

Trois (3) décalages :

Période de calage :

Table 4.29 : Résultats de prévision des apports journaliers avec trois décalages (calage)

variante	AJ2N4D30	AJ2N4D60	AJ2N4D80	AJ5N4D30	AJ5N4D60	AJ5N4D80
RMSE	0.7018	0.7987	0.8212	0.7104	0.7760	0.3701
r^2	0.4204	0.3345	0.3712	0.4067	0.3715	0.8204

AJ10N4D30	AJ10N4D60	AJ10N4D80
0.7010	0.7687	0.8184
0.4229	0.3831	0.3745

Période de validation :

Table 4.30 : Résultats de prévision des apports journaliers avec trois décalages (validation)

variante	AJ2N4D30	AJ2N4D60	AJ2N4D80	AJ5N4D30	AJ5N4D60	AJ5N4D80
RMSE	1.0380	1.1524	1.3228	1.0430	1.1689	1.3041
r^2	0.3235	0.3351	0.3236	0.3131	0.3148	0.3518

AJ10N4D30	AJ10N4D60	AJ10N4D80
1.0431	1.1816	1.3184
0.3187	0.2912	0.3324

8.2. Prévisions journalières des Volumes:

Un (1) décalage :

Période calage :

Table 4.31: Résultats de prévision des volumes journaliers avec un décalage (calage)

variante	VJ2N2D30	V2N2D60	VJ2N2D80	VJ5N2D30	VJ5N2D60	VJ5N2D80
RMSE	1.4097	1.4033	1.4390	1.4155	1.4050	1.4379
r^2	0.9986	0.9988	0.9988	0.9986	0.9988	0.9988

VJ10N2D30	VJ10N2D60	VJ10N2D80
1.4086	1.4056	1.4362
0.9986	0.9988	0.9988

Période validation :

Table 4.31 : Résultats de prévision des volumes journaliers avec un décalage (validation)

variante	VJ2N2D30	VJ2N2D60	VJ2N2D80	VJ5N2D30	VJ5N2D60	VJ5N2D80
RMSE	1.3257	1.2549	0.9029	1.3627	1.2556	0.9036
r^2	0.9987	0.9986	0.9988	0.9986	0.9986	0.9988

VJ10N2D30	VJ10N2D60	VJ10N2D80
1.3356	1.2555	0.9005
0.9987	0.9986	0.9988

Trois (3) décalages :

Période de calage :

Table 4.32: Résultats de prévision des volumes journaliers avec trois décalages (calage)

Variante	VJ2N4D30	VJ2N4D60	VJ2N4D80	VJ5N4D30	VJ5N4D60	VJ5N4D80
RMSE	0.9990	1.2323	1.3510	1.2397	1.2383	1.2903
r²	0.9987	0.9991	0.9989	0.9990	0.9991	0.9990

VJ10N4D30	VJ10N4D60	VJ10N4D80
1.1951	1.2323	1.3510
0.9990	0.9991	0.9989

Période de validation :

Table 4.33: Résultats de prévision des volumes journaliers avec un décalage (validation)

variante	VJ2N4D30	VJ2N4D60	VJ2N4D80	VJ5N4D30	VJ5N4D60	VJ5N4D80
RMSE	1.3458	1.2016	0.8136	1.2595	1.2230	0.7968
r²	0.9987	0.9987	0.9990	0.9988	0.9987	0.9991

VJ10N4D30	VJ10N4D60	VJ10N4D80
1.4155	1.1237	0.8057
0.9985	0.9989	0.9991

8.3. Prévisions journalières des lames évaporées :

Un (1) décalage :

Période calage :

Table 4.34: Résultats de prévision des lames évaporées journalières avec un décalage (calage)

variante	LJ2N2D30	LJ2N2D60	LJ2N2D80	LJ5N2D30	LJ5N2D60	LJ5N2D80
RMSE	0.8155	1.0419	0.9785	1.1858	1.0407	0.9794
r^2	0.7278	0.7606	0.7755	0.7295	0.7613	0.7752

LJ10N2D30	LJ10N2D60	LJ10N2D80
1.1835	1.0420	0.7763
0.7304	0.7606	0.8032

Période validation :

Table 4.35 : Résultats de prévision des lames évaporées journalières avec un décalage (calage)

variante	LJ2N2D30	LJ2N2D60	LJ2N2D80	LJ5N2D30	LJ5N2D60	LJ5N2D80
RMSE	0.8155	0.7523	0.7446	0.8170	0.7527	0.7444
r^2	0.8085	0.8202	0.8032	0.8083	0.8195	0.8029

LJ10N2D30	LJ10N2D60	LJ10N2D80
0.8121	0.7538	0.7428
0.8096	0.8187	0.8032

Trois (3) décalages :

Période de calage :

Table 4.36 : Résultats de prévision des lames évaporées journalières avec trois décalages (calage)

variante	LJ2N4D30	LJ2N4D60	LJ2N4D80	LJ5N4D30	LJ5N4D60	LJ5N4D80
RMSE	1.0688	0.9537	0.9044	1.0635	0.9435	0.9026
r^2	0.7802	0.7994	0.8082	0.7826	0.8037	0.8090

LJ10N4D30	LJ10N4D60	LJ10N4D80
1.0704	0.9517	0.8962
0.7795	0.8006	0.8117

Période de validation :

Table 4.37 : Résultats de prévision des lames évaporées journalières avec trois décalages (calage)

variante	LJ2N4D30	LJ2N4D60	LJ2N4D80	LJ5N4D30	LJ5N4D60	LJ5N4D80
RMSE	0.7764	0.7267	0.7135	0.7842	0.7236	0.7119
r^2	0.8250	0.8312	0.8182	0.8216	0.8333	0.8190

LJ10N4D30	LJ10N4D60	LJ10N4D80
0.7767	0.7282	0.7119
0.8244	0.8323	0.8196

9. Prévisions journalières par la logique floue :

9.1. Prévisions des apports :

Un (1) décalage :

Période calage :

Table 4.38 : Résultats de prévisions la logique floue des apports journaliers avec un décalage (calage)

variante	AJ2RF2D60
RMSE	0.7744
r^2	0.3733

Période validation :

Table 4.39 : Résultats de prévision des apports journaliers par la logique floue avec un décalage (validation)

variante	LM2RF2D60
RMSE	1.1306
r^2	0.3509

Trois (3) décalages :

Période calage :

Table 4.40 : Résultats de prévision apports journaliers par la logique floue avec trois décalages (calage)

Variante	AJ2RF4D60
RMSE	0.7587
r^2	0.0255

Période validation :

Table 4.41 : Résultats de prévision des apports journaliers par la logique floue avec trois décalages (validation)

variante	AJ2RF4D60
RMSE	8.0239
r^2	0.3984

9.2. Prévisions journalières des lames évaporées :

Un (1) décalage :

Période calage :

Table 4.42: Résultats de prévisions la logique floue des lames évaporées avec un décalage (calage)

variante	LJ2RF2D60
RMSE	1.0474
r^2	0.7581

Période validation :

Table 4.43 : Résultats de prévision des lames évaporées journalières par la logique floue avec un décalage (calage)

variante	LJ2RF2D60
RMSE	0.7539
r^2	0.8192

Trois (3) décalages :

Période calage :

Table 4.44 : Résultats de prévision des lames évaporées journalières par la logique floue avec trois décalages (calage)

variante	LJ2RF4D60
RMSE	0.9391
r^2	0.8055

Période validation :

Table 4.45 : Résultats de prévision des volumes journaliers par la logique floue avec trois décalages (validation)

variante	LJ2RF4D60
RMSE	0.7234
r^2	0.8330

9.3. Prévisions journalières des volumes

Un (1) décalage :

Période calage :

Table 4.46 : Résultats de prévision des volumes journaliers par la logique floue avec un décalage (calage)

variante	V2RF2D30
RMSE	1.4098
r^2	0.9986

Période validation :

Table 4.47 : Résultats de prévision des volumes journaliers par la logique floue avec un décalage (validation)

variante	V2RF2D30
RMSE	1.3200
r^2	0.9987

Trois (3) décalages :

Période calage :

Table 4.48 : Résultats de prévision des volumes journaliers par la logique floue avec trois décalages (calage)

variante	V2RF4D30
RMSE	1.0493
r^2	0.9992

Période validation :

Table 4.49 : Résultats de prévision des volumes journaliers par la logique floue avec un décalage (validation)

variante	V2RF4D30
RMSE	1.6011
r^2	0.9981

Conclusion :

L'approche journalière a été développée de façon à permettre au gestionnaire d'avoir une vision à très court terme pour la gestion des barrages réservoirs, ce pas de temps permet d'apporter les correctifs nécessaires dans le cas où il y'a un changement dans le pas de temps mensuel. Et effectivement Une amélioration des résultats est constatée lors de la prévision en pas du temps journalier en La comparant par celle au pas du temps mensuel.

10. Prévision avec et sans correction des volumes

À l'addition de la prévision mensuelle des volumes, nous avons appliqué deux méthodes de prévisions : prévisions avec correction et prévisions sans correction avec quatre décalages. Et ceci est pour les deux années 2014 et 2015 ;

10.1. Prévision sans correction:

Pour prévoir la valeur au temps $t+2$, on garde la valeur obtenue au temps $t+1$ par le modèle de prévision,

Pour prévoir la valeur au temps $t+3$, on garde les valeurs calculées par le modèle au temps $t+1$, et $t+2$, et ainsi de suite jusqu'à la fin de la prévision.

Les résultats ainsi trouvés sont présentés par le tableau 4.50 :

Table 4.50 : Résultats d'une Prévision sans correction des volumes

Volume observé	Volume prévu
78.842	102.544
71.664	64.542
72.546	67.133
78.563	88.224
79.356	83.208
96.000	79.587
130.393	116.014
142.069	157.312
136.923	148.120
128.462	129.319
116.912	117.730
106.758	120.025
98.139	109.574
91.433	98.552
85.647	94.323
87.431	90.327
112.384	82.239
135.118	83.957
142.340	120.639
144.710	141.623
139.917	135.801
132.356	133.902
115.181	121.813
104.881	105.859

10.2. Prévision avec correction:

Pour prévoir la valeur au temps $t+2$, on ne garde pas la valeur obtenue au temps $t+1$ par le modèle de prévision, mais on va la remplacer par la valeur observée qui sera connue à $t+1$.

Pour prévoir la valeur au temps $t+3$, on ne garde pas les valeurs calculées par le modèle au temps $t+1$, et $t+2$, mais on va les remplacer par les valeurs observées au temps $t+1$, et $t+2$, et ainsi de suite jusqu'à la fin de la prévision.

Les résultats ainsi trouvés sont présentés par le tableau **4.51**:

Table IV.51: Résultats d'une prévision avec correction des volumes

Volume observé	Volume prévu
78.842	102.544
71.664	104.528
72.546	107.422
78.563	116.002
79.356	117.894
96.000	122.227
130.393	124.162
142.069	121.439
136.923	112.915
128.462	103.668
116.912	100.781
106.758	97.170
98.139	72.157
91.433	73.571
85.647	80.744
87.431	79.918
112.384	89.489
135.118	97.895
142.340	104.996
144.710	111.387
139.917	119.790
132.356	121.744
115.181	125.247
104.881	118.617

Conclusion Générale

Le travail présenté dans ce mémoire de master est une suite du projet fin d'étude, donc après la gestion par les algorithmes génétique et la simulation par le modèle GESTOP on vient à la prévision par les réseaux de neurones et la logique floue en pas du temps mensuel et journalier; lorsque nous avons entamé ce projet, l'objectif principale était de développer une approche pour la prévision des apports vu leur comportement aléatoire difficile à prévoir, voire impossible, et non pas seulement de développer de procéder la prévision moyennant l'intelligence artificielle connue par sa robustesse et efficacité par rapports aux techniques traditionnelles.

Au pas de temps mensuel, nous avons une vision assez lointaine de l'évolution de la gestion des barrages réservoirs ; pour les variables lames évaporées et volumes d'eau, les résultats sont assez probants, la difficulté réside dans la prévision de la variable apports, parce qu'elle est très difficile, voire impossible à prévoir du fait que sa structure aléatoire.

Pour que le gestionnaire puisse avoir une vision à très court terme, nous avons procédé à une prévision au pas de temps journalier des trois paramètres.

Donc l'approche journalière a été développée de façon à permettre au gestionnaire d'avoir une vision à très court terme pour la gestion des barrages réservoirs,

Les réseaux de neurones et la logique floue donne des résultats similaires, ça veut dire qu'ils arrivent à prévoir les mêmes variables avec des taux de réussite assez proche les uns des autres, et lorsque la complexité du phénomène devient plus grande, c'est à dire la variable est aléatoire les résultats ne sont pas bon pour les deux techniques.

Pour pouvoir exploiter l'équation de continuité dans le futur, il faut que les deux variables prépondérantes du système à savoir les apports et les lames évaporées doivent être bien prévues, chose qu'on n'a pas pu atteindre, et c'est pour cela qu'on s'est focalisé sur la variable volume du réservoir qui est la résultante de plusieurs paramètres aléatoires et cycliques qui s'intègre dans le système de gestion.

Et là les résultats obtenus par les deux techniques de prévision à savoir les réseaux de neurones et la loque floue ce sont avéré être excellent, ce qui permet une bonne lisibilité de la gestion futur du barrage réservoir.

Bibliographie

ANDRE, J.C., CANEILL, J.Y., DEQUE, M., ROGEL, P., TERRAY, L., TOURRE, Y., (2002) : La prévision du climat, de l'échelle saisonnière à l'échelle décennale. *Comptes Rendus Géosciences*, Vol., 334(16), pp. 1115-1127.

Andréassian V., Michel C., Perrin C., (2007) : Modèles hydrologiques du Génie Rural (GR), Cemagref,

Arnold J G., Bigner R L., Harmel R D., Moriasi D N., Van Liew M W., Veith T L., (2007) : Model Evaluation Guidelines for Systematic Quantification of Accuracy in Watershed Simulation, *American Society of Agricultural and Biological Engineers ISSN 0001-23351*, Vol. 50(3): 885-900.

BARDOSSY, A., (1990) : Note on fuzzy regression. *Fuzzy sets and systems*, 37, p 65-75.

BECKIE, R., (1996) : Measurement scale, network sampling scale, and groundwater model parameter. *Water Resources Research*, 32(1), pp. 65-76.

Benkaci T., (2006) : Modélisation pluie-débit mensuelle et journalière par les modèles conceptuels et les systèmes neuro-flous (application aux bassins algériens), Thèse de doctorat, Institut National Agronomique d'Alger, 2006.

DAVALO, E., NAÏM, P., (1993) : Des réseaux de Neurones, Edition Eyrolles, Paris.

Croke B.W & Jakeman A.J. (2004) A catchment moisture deficit module for the IHACRES rainfall-runoff model. *Environmental Modelling & Software*, 19; 1-5.

Cybenko G. (1989) Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signal and Systems*, 2(4):303-314, 1989.

Dechemi N. & Chambaz H. (1994) La prévision des crues du bassin versant de l'oued Dis (Sebbaou) par la méthode DPFT. *Revue Sciences de l'eau*, 7; 331-342.

Dechemi N., Bermad A., Tachet K. (1997) Etude et simulation de l'évapotranspiration (ETP) par l'analyse en composantes principales (ACP). *ICID Journal*, 46; 45-60.

Dechemi N., Benkaci Ali T., Issolah A. (2003) Modélisation des débits mensuels par les modèles conceptuels et les systèmes neuro-flous. *Revue Sciences de l'eau*, 3; 407-425.

Delattre, P. (1979) Le problème de la justification des modèles dans le cadre du formalisme des systèmes de transformation. Actes du colloque 'Elaboration et Justification des Modèles', Tome I, Ed Maloine; 97-128.

FIORDALISO, A., (1999) : Systèmes flous et prévision de séries temporelles. Edition HERMES.

FRENCH, M.N., KRAJEWSKI, W.F., CUYKENDALL, R.R., (1992) : Rainfall Forecasting in Space and Time using a Neural Network, *Journal of Hydrology*, Vol. 137, p. 1-31.

JODOUIN, J. F., (1994) : Les réseaux de neurones, Edition Hermès, Paris.

Haziza E., (2003) : Modélisation mensuelle pluie-débit /apports de la spatialisation-cas des données de sols, DEA en science de l'eau et de l'environnement, Université de Montpellier II.

Heddiam, S. (2012) « thèse de doctorat : Contribution à la modélisation de la qualité des eaux approche par les réseaux de neurones, les systèmes neuroflous et les méthodes de classification Application à la station de traitement des eaux de BOUDOUAOU-ALGERIE

Hopfield J.J. (1982) Neural networks and physical systems with emergent collective computational abilities," Proceedings of the National Academy of Sciences, **79**; 2554-2558.

Hreiche A.,(2003). Modélisation conceptuelle de la transformation pluie-débit dans le contexte méditerranéen, Thèse de doctorat, Université Montpellier et Université Saint-Joseph,

MINNS, A., HALL, J., (1996): Artificial neural network as rainfall-runoff models. Journal des sciences hydrologiques, 41(3), pp. 399-417.

MIQUEL, J., (2001) : Hydrologie Statistique-introduction à l'étude des processus hydrométéorologiques. Application à la prédétermination des débits de crue. ENPC, Paris.

Mamdani E.H. (1974) Application of Fuzzy Algorithms for Control of a simple Dynamic Plant, Proc. of the IEE Control and Science, **121**; (12), 1585-1588.

Mamdani E.H. Assilian S. (1975) An experiment in Linguistic Synthesis with a fuzzy logic Controller, Int. Journal of Man-Machines Studies, **7**; 1-13.

Marsi S.F., Bkey G.A., Safford F. (1976) An adaptive random search method for identification of large scale nonlinear systems. Proceeding of the IFAC Symposium, 1645-1654.

Perrin C., (2000) : Vers une amélioration d'un modèle global pluie-débit au travers d'une approche comparative, Thèse de doctorat, de l'INPG, Institut National Polytechnique de Grenoble, Octobre

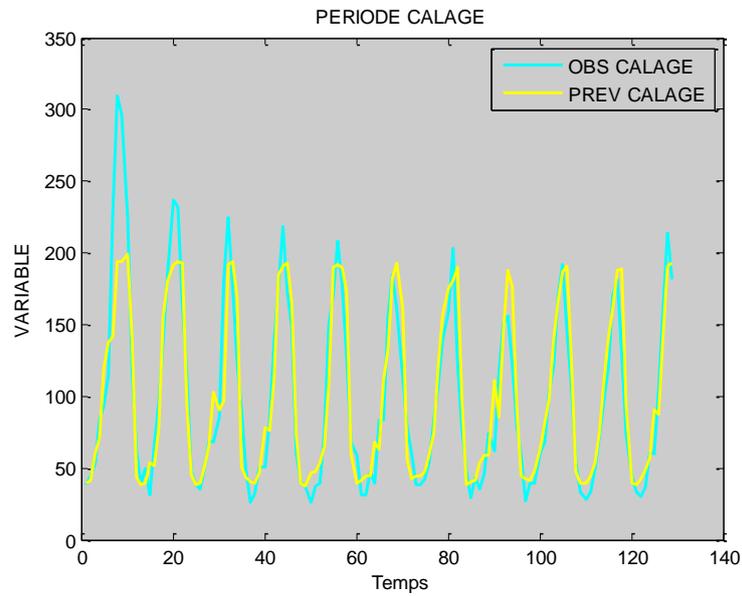
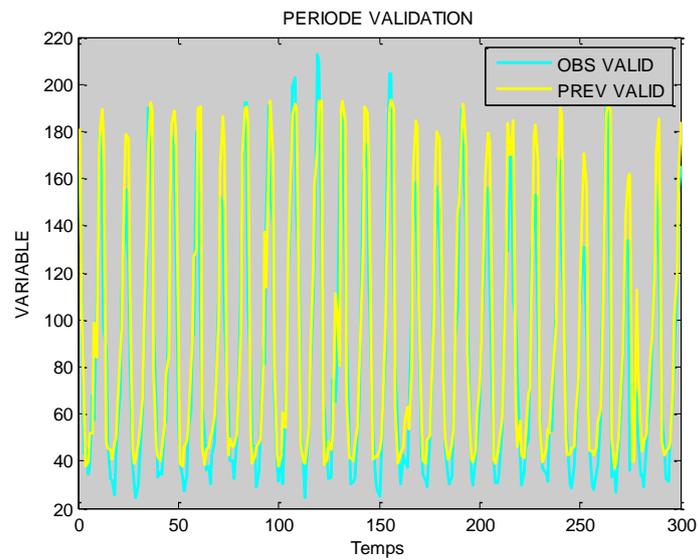
Perrin C., Michel C., Andréassian V., (2007) :Modèles hydrologiques du Génie Rural (GR), Cemagref,.

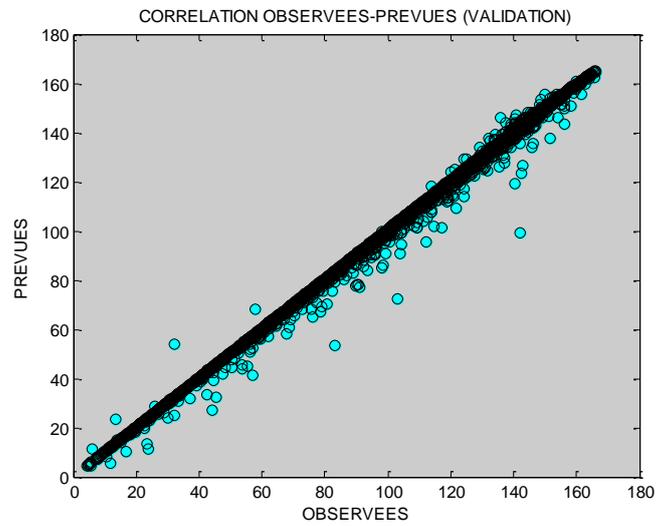
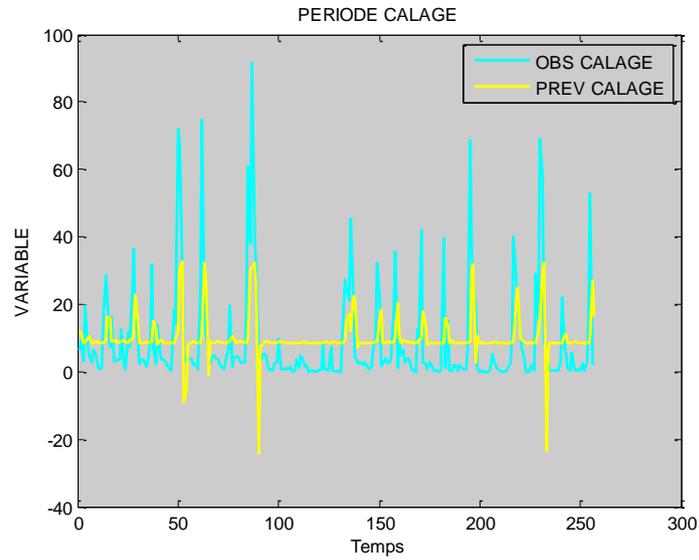
Sétra,(2010) Calage et validation des modèles de trafic, Juillet 2010, Service d'études sur les transports, les routes et leurs aménagements.

Souag.D (2007) :Développement d'outils pour la gestion des barrages réservoirs basés sur la simulation et la prévision des paramètres hydrométéorologiques.Thèse de doctorat.Université des Sciences et de la Technologie Houari Boumediene Faculté de Génie Civil.

Vasile O E., (2008) : Contribution au pronostic de défaillances par réseau neuro-flou: Maîtrise de l'erreur de prédiction, Thèse de doctorat en automatique, Ecole Doctorale Sciences Physiques pour l'Ingénieur et Microtechniques.

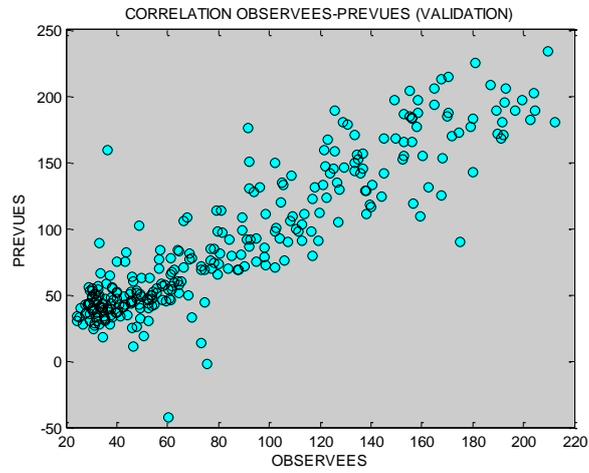
Annexes

Annexe 1 : Prédiction mensuelle par les réseaux de neurones :LM2N4D30**Annexe 1-1 : Comparaison entre les variables observées et prévues des lames évaporées en phase calage****Annexe 1-2 : Comparaison entre les variables observées et prévues des lames évaporées en phase validation**

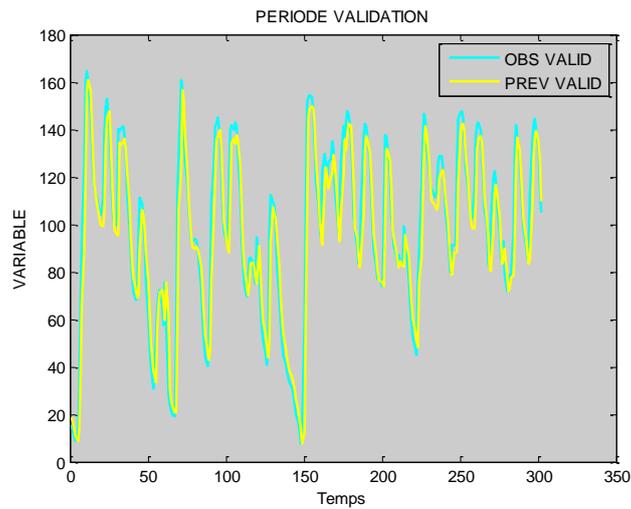
Annexe 1-3 : Corrélation entre volumes observées et prévues en phase validation, VM2N2D30**Annexe 1-4 : Comparaison entre les apports observés et prévus en phase calage : AM2N4D60**

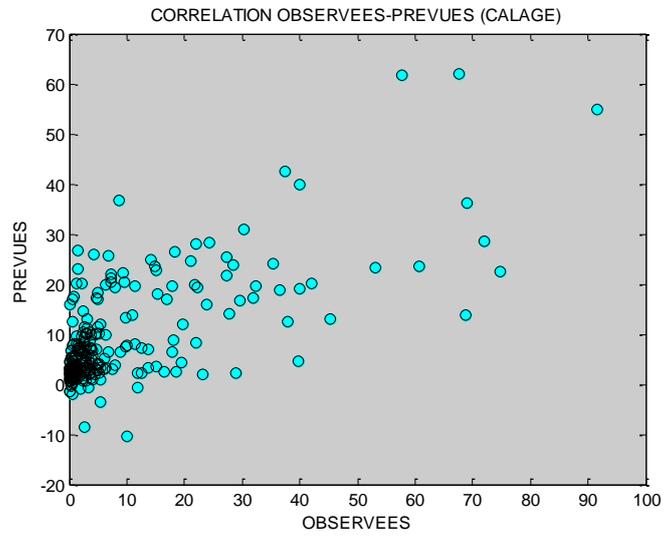
Annexe 2 : Prédiction mensuelle par la logique floue

Annexe 2-1 : Corrélation entre les lames évaporées observées et prévues en phase validation LM2RF4D30



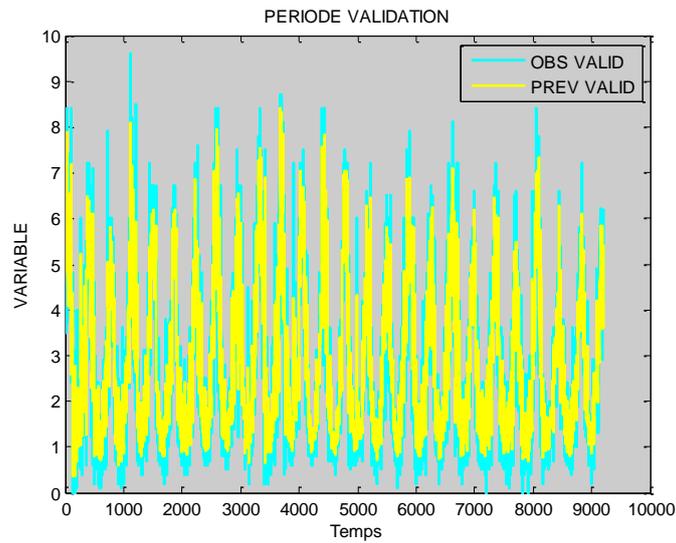
Annexe 2-2 : Comparaison entre les volumes observés et prévus en phase validation VM2RF2D30



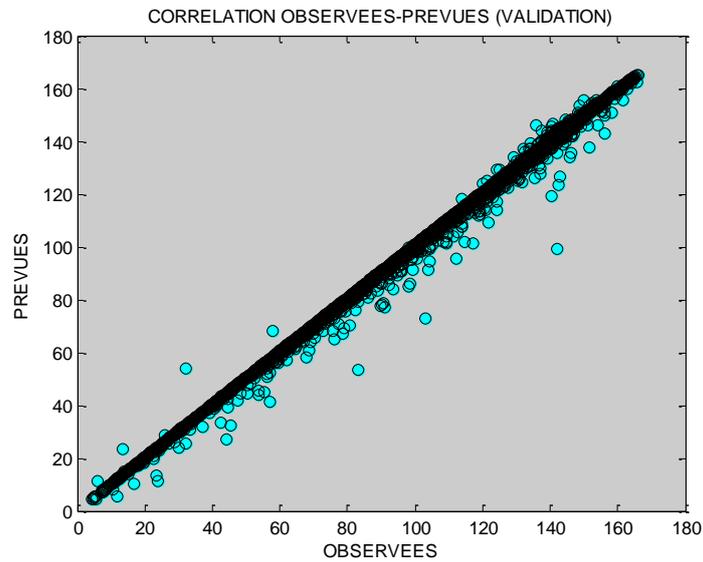
Annexe 2-3 : Corrélation entre les apports observés et prévus en phase calage AM2RF4D60

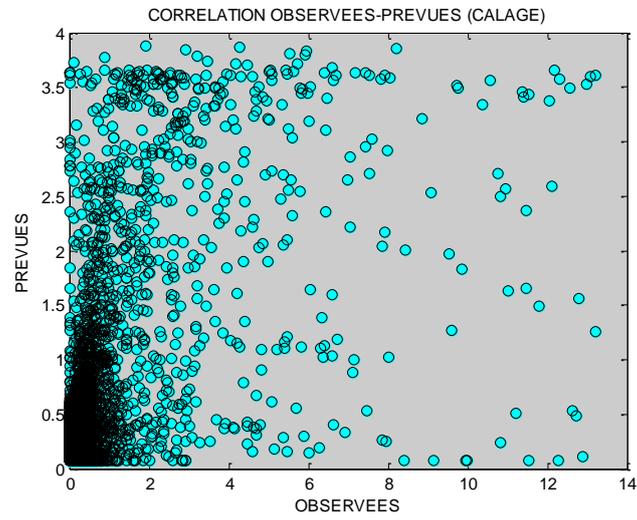
Annexe 3: Prédiction journalière par les réseaux de neurones

Annexe 3-1: Comparaison entre les lames évaporées observées et prévues en phase validation : LJ2N4D60



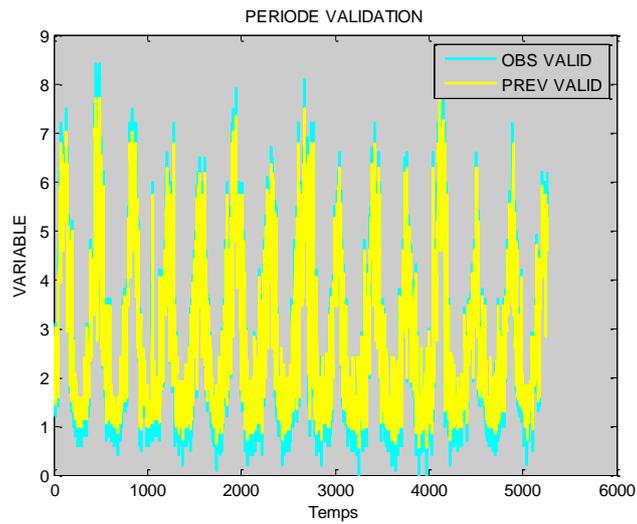
Annexe 3-2: : Corrélation entre les volumes observés et prévus en phase validation : VJ2N2D30



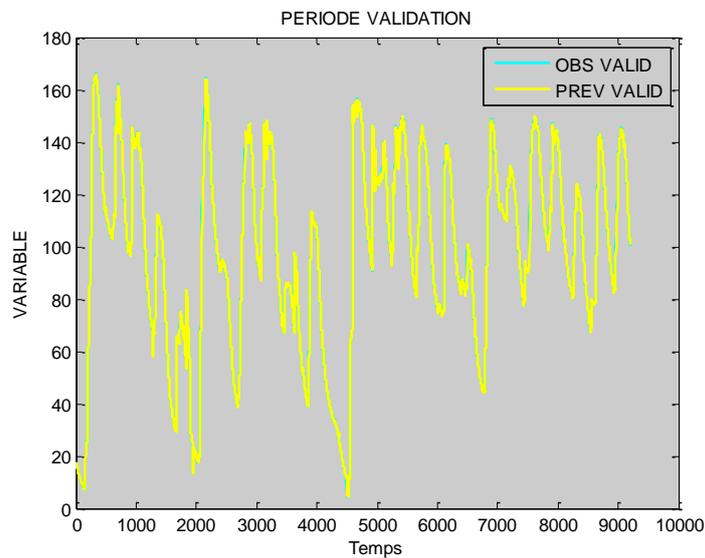
Annexe 3-2: : Corrélation entre les apports observés et prévus en phase calage AJ2N2D30

Annexe 4: Prédiction journalière par la logique floue

Annexe 4-1: Comparaison entre les lames évaporées observées et prévues en phase validation LJ2RF2D60



Annexe 4-2: Comparaison entre les volumes observés et prévus en phase validation VJ2RF2D30



Annexe 4-3: Comparaison entre les apports observés et prévus en phase calage AJ2RF2D60