

Etude des effets des algorithmes d'apprentissage et des fonctions de transfert sur la performance des modèles statistiques neuronaux : Application dans le domaine météorologique

Hicham El Badaoui¹, Abdelaziz Abdallaoui², Samira Chabaa³

^{1,2} Université Moulay Ismail, Faculté des Sciences, Laboratoire de Chimie-Biologie Appliquées à l'Environnement, Equipe Chimie Analytique et Environnement, B.P. 11201, Zitoune, Meknès, Maroc.

³ Université Ibn Zohr, Ecole Nationale des Sciences Appliquées, Département Génie industriel BP. 1136, Agadir, Maroc.

Résumé:- Ce travail présente le développement d'un modèle de réseau de neurones artificiel (RNA) de type Perceptron Multicouche (PMC) pour la prédiction du taux d'humidité de la zone de Chefchaouen. Notre objectif c'est traiter une série chronologique des données mesurées entre les années 2008 et 2013 pour évaluer la réponse du réseau. Cette série est constituée d'un certain nombre de paramètres météorologiques telles que le taux d'humidité, la température de l'air, la température de rosée, la pression atmosphérique, la visibilité, la nébulosité, la vitesse du vent et la précipitation. Pour déterminer l'architecture du réseau à utiliser, nous avons varié le nombre des couches cachées, le nombre de neurones dans la couche cachée, les fonctions de transfert et les couples de fonctions de transfert ainsi que les algorithmes d'apprentissage. Plusieurs critères statistiques comme l'erreur quadratique moyenne (MSE) et le coefficient de corrélation (R), ont été étudiés pour l'évaluation des modèles développés. Les meilleurs résultats ont été obtenus avec un réseau d'architecture [7-5-1], les fonctions d'activations (Tansig-Purelin) et un algorithme d'apprentissage de Levenberg-Marquardt.

Mots clés:- Réseau de neurones artificiel, Perceptron Multicouche, Prédiction, Algorithme d'Apprentissage, Humidité.

I. INTRODUCTION

La météorologie accumule des quantités énormes de données. Ces données proviennent d'observations effectuées au sol, en altitude, en mer et dans l'espace. Elles sont hétérogènes et fortement bruitées. Pour extraire des informations précises de ces données, les météorologues utilisent différents types de modèles qui dépendent des domaines comme la physique ou la statistique. La physique a pour objet la représentation de la réalité à l'aide de modèles qualifiés de "boite blanche". Tandis que la statistique essaie de construire ces modèles mathématiques de type "boite noire" à partir des données empiriques. L'information extraite va dépendre de l'ensemble des modèles physiques ou statistiques utilisés pour décrire le problème et les différentes hypothèses qui seront choisies au départ. Dans le domaine météorologique, les lois des paramètres climatiques sont non linéaires et afin de modéliser ce type de problèmes, nous nous sommes intéressés particulièrement à un modèle typique de réseau neuronal connu sous le nom Perceptron Multicouche. Les réseaux de neurones artificiels sont extrêmement appliqués pour résoudre des phénomènes fortement complexes et non linéaires. Dans la littérature, les réseaux de neurones artificiels ont présenté un grand succès dans l'analyse statistique, les traitements de données et la prévision des paramètres environnementaux, et différents travaux de recherches ont été proposés dans le cadre de la prédiction en utilisant les réseaux de neurones de type PMC.

Abdelli S. et al., en 2012, [1] ont présenté la méthode des réseaux de neurones de type perceptron multicouche, pour prédire des polluants d'air. Leur objectif est de prévoir les concentrations de NO_2^- et de trouver ses relations avec les autres variables météorologiques.

Fock E. et al., en 2000, [2] ont traité un modèle statistique neuronal pour la modélisation thermique des bâtiments. Cette démarche basée sur la prédiction de la température de surface intérieure d'une paroi montre la faisabilité et la possible implémentation du réseau dans un code.

Abdallaoui A. et al., 2013, [3] ont utilisé aussi les réseaux de neurones multicouche pour la prédiction des concentrations des métaux lourds dans les sédiments fluviaux marocains, à partir d'un certain nombre de paramètres physico-chimiques.

L'humidité relative est l'un des paramètres météorologiques importants vue son influence sur la santé de l'homme, les matières de construction, l'agriculture et l'environnement, etc. Ainsi nous avons jugé utile de développer des modèles mathématiques statistiques neuronaux pour la prédiction de l'humidité relative dans la zone de Chefchaouen à partir d'un certain nombre de paramètres météorologiques.

II. MATÉRIEL ET MÉTHODES

A. Présentation de la zone d'étude

Notre zone d'étude est la province de Chefchaouen qui est localisée dans une région importante caractérisée par une grande diversité des sols et des formations géologiques. Elle se trouve dans la zone de transition entre le domaine des nappes de flyschs et le domaine interne.



Fig.1 : Situation géographique de la zone de Chefchaouen (Maroc).

La Province de Chefchaouen, située au Nord-Ouest du Maroc (Latitude: 35°10'17 nord; Longitude: 5° 16 '11 Ouest; Altitude: 564 mètres) sur la chaîne rifaine, est limitée au Nord par la méditerranée sur une longueur de 120 Km, au Sud par les provinces de Taounate et Sidi Kacem, à l'Est par la province d'Al Hoceima et à l'Ouest par les provinces de Tétouan et Larache. C'est une province qui relève de la région administrative de Tanger-Tétouan (Fig. 1).

B. Base de données

La base de données utilisée dans le cadre de cette étude est constituée par les valeurs de huit variables météorologiques relatives à la zone de Chefchaouen:

- Sept variables indépendantes (explicatives) : Température de l'air, température de rosée, pression atmosphérique, visibilité, nébulosité, vitesse du vent et la précipitation.
- Une variable dépendante (à prédire) : le taux d'humidité (l'humidité relative).

Ces valeurs ont été relevées toutes les quatre heures pendant 1856 jours entre les années 2008 et 2013. Elles ont été converties en moyennes journalières pour toutes les variables, sauf la quantité de précipitations qui a été transformée en une valeur cumulative de la journée [4].

C. Mise en forme des données

Les données d'entrée (variables indépendantes) sont des valeurs brutes non transformées qui ont des ordres de grandeurs très différents selon les variables. Afin d'uniformiser les échelles de mesures, ces données sont converties en variables standardisées. En effet, les valeurs I_i de chaque variable indépendante (i) ont été standardisées par rapport à sa moyenne et son écart-type suivant la relation :

$$\hat{I}_i = \frac{I_i - \mu_i}{\sigma_i}$$

\hat{I}_i : Valeur standardisée relative à la variable i ;

I_i : Valeur brute relative à la variable i ;

μ_i : Valeur moyenne relative à la variable i ;

σ_i : écart-type relative à la variable i .

Les valeurs correspondantes aux variables dépendantes ont été également normalisées dans l'intervalle [-1,1] à partir de la relation suivante [5]:

$$\hat{I}_i = \frac{2(I_i - \text{Min}(I_i))}{(\text{Max}(I_i) - \text{Min}(I_i))} - 1$$

III. RESEAUX DE NEURONES ARTIFICIELS (RNA)

Un réseau de neurones artificiels est un modèle de calcul dont l'inspiration originelle était un modèle biologique, c'est-à-dire le modèle du cerveau nerveux humain. Les réseaux de neurones artificiels sont optimisés par des calculs d'apprentissage de type statistique. Ils sont placés d'une part, dans la famille des applications statistiques, permettant de générer de vastes espaces fonctionnels, souples et partiellement structurés, et d'autre part dans la famille des méthodes de l'intelligence artificielle, permettant de prendre des décisions s'appuyant davantage sur la perception que sur le raisonnement logique formel [6].

A. Modèle mathématique d'un neurone formel

Le modèle mathématique d'un neurone artificiel est illustré par la figure 2. Un neurone est essentiellement constitué d'un intégrateur qui effectue la somme pondérée de ses entrées. Le résultat de cette somme ensuite transformée par une fonction de transfert f qui produit la sortie O_n du neurone. Les n entrées du neurone correspondent au vecteur noté: $\vec{I} = (I_1, I_2, I_3, \dots, I_n)$ et le vecteur de poids est représenté par : $\vec{W} = (W_1, W_2, W_3, \dots, W_n)$.

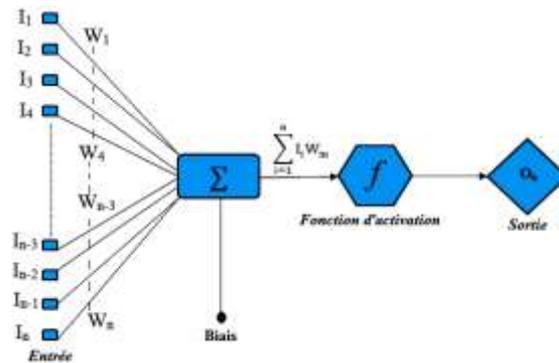


Fig.2 : Structure d'un neurone artificiel

La sortie O_n de l'intégrateur est définie par l'équation suivante : $O = W^T I - b$

Le résultat de la somme pondérée s'appelle le niveau d'activation du neurone. Le biais b s'appelle aussi le seuil d'activation du neurone [7].

B. Choix d'un réseau de neurones

Il existe un grand nombre de types de réseaux de neurones, dont chacun a des avantages et des inconvénients. Le réseau choisi dans notre étude est un réseau perceptron multicouche (PMC). Ce choix est fait pour la facilité et la rapidité de sa construction et encore par le fait que notre problème ne présente qu'un nombre considéré limité de variables d'entrées [8]. La figure 3 illustre le perceptron multicouche dans le cas d'une seule couche d'entrée, une seule couche cachée et une seule sortie.

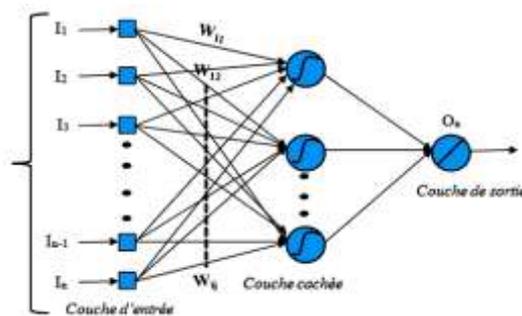


Fig.3 : Exemple d'architecture d'un PMC à n variables d'entrée, n neurones dans la couche cachée et une seule sortie

Le perceptron multicouche est composé d'un assemblage de neurones répartis sur plusieurs couches. La première couche correspondant au vecteur composé des données d'entrée et la dernière couche au vecteur de

sortie composé des valeurs désirées entre les deux couches. Entre ces couches il existe un certain nombre de couches cachées. La complexité du réseau est en fonction du nombre de ces couches cachées et du nombre des neurones élémentaires composant chaque couche.

C. Indicateurs statistiques

Pour évaluer les performances des différents modèles développés, nous avons utilisé deux indicateurs statistiques :

- Le coefficient de corrélation R donné par la relation [9]:

$$R = \frac{\sqrt{\sum_i (\hat{O}_i - \bar{O})^2}}{\sqrt{\sum_i (O_i - \bar{O})^2}} = \sqrt{1 - \frac{\sum_i (O_i - \hat{O}_i)^2}{\sum_i (O_i - \bar{O})^2}}$$

$\sum_i (\hat{O}_i - \bar{O})^2$: Somme des carrés expliquée par le modèle ;

$\sum_i (O_i - \bar{O})^2$: Somme des carrés totale.

- L'erreur quadratique moyenne MSE : elle représente la moyenne arithmétique des carrés des écarts entre les prévisions et les observations [10].

$$MSE = \frac{1}{N} \sum_1^N (\hat{O}_i - O_i)^2$$

IV. RESULTAT ET DISCUSSION

A. Régression linéaire multiple (RLM)

La régression linéaire multiple est utilisée pour prévoir les valeurs d'une variable dépendante (le taux d'humidité) à partir de variables explicatives (paramètres météorologiques). L'équation du modèle est la suivante :

$$H (\%) = 71,187 - 1,995 \times T_a + 1,719 \times T_r + 1,915 \times P_a - 0,564 \times Vis + 0,927 \times Néb + 0,023 \times P_r - 0,04 \times V$$

$$N=1856 \text{ Jours ; } R= 0,865 ; MSE = 0,0260$$

Avec :

T_r : Température rosée ;

P_a : Pression atmosphérique ;

Vis : Visibilité ;

$Néb$: Nébulosité ;

P_r : Précipitations ;

V : Vitesse du vent ;

T_{air} : Température de l'air ;

H : Humidité relative.

Le coefficient de corrélation obtenu par le modèle (RLM) est $R=0,865$ et l'erreur quadratique moyenne est $MSE = 0,026$, ceci montre que l'humidité relative n'est pas corrélé linéairement avec les autres paramètres météorologiques.

Avec ce modèle, il est difficile d'imaginer que l'on puisse établir un modèle entièrement déterministe qui fournirait une prédiction vraiment exacte. Pour dépasser ces difficultés, nous avons développé des modèles mathématiques de type « boîte noire » à savoir : les réseaux de neurones artificiels (RNA).

B. Réseaux de neurones artificiels (RNA)

Dans le cadre de ce travail, nous avons utilisé les réseaux de neurones non récurrents de type PMC. Le choix d'exploiter un type de réseaux de neurones et écarter un autre n'est pas arbitraire, mais repose sur les fondements qu'exigent le problème en question, sachant que chaque type de réseaux, en termes d'apprentissage, est conseillé pour une application donnée. Nous avons opté donc pour le perceptron multicouche puisqu'il répond aux spécificités du problème de prédiction. Ce type de réseau apporte des résultats très significatifs et le taux de précision de la prédiction étant meilleur. Les algorithmes d'apprentissage présentés par la suite font partie de la boîte à outils du logiciel MATLAB des réseaux de neurones. Pour déterminer l'architecture du réseau à utiliser, nous avons varié le nombre de couches cachées, le nombre de neurones dans une couche

cachée, les fonctions de transfert, le nombre d'itération, le pas d'apprentissage et les algorithmes d'apprentissage [11]. Dans le cadre de cette étude, nous avons divisé, aléatoirement notre base de données en trois parties : 60% pour l'apprentissage, 20% pour le test et 20% pour la validation.

✓ **Le nombre de couches cachées :**

Selon l'architecture choisie, tous les neurones ou une partie des neurones dans une couche sont connectés avec tous ou une partie des neurones des couches adjacentes. Le nombre de couches cachées et le nombre de neurones par couche dépend du modèle spécifique, de la vitesse de convergence, des capacités de généralisation, du processus physique et des données d'apprentissage [12]. Dans tout notre travail, nous sommes contents d'une seule couche cachée, car nous avons remarqué que l'augmentation du nombre de couches cachées augmente la charge des calculs sans aucun gain de performance.

✓ **Les algorithmes d'apprentissage :**

Dans cette étude nous avons utilisé quatre algorithmes d'apprentissage, qui sont qualifiés de haute performance :

- Resilient Back Propagation (RBP);
- Gradient Descent Back Propagation (GD) ;
- Gradient Conjugué Scalaire (GCS);
- Levenberg-Marquardt (LM).

Pour chaque algorithme d'apprentissage, nous avons changé le nombre de neurones dans la couche cachée ainsi que les couples de fonctions de transfert. Les performances ont été évaluées grâce à l'erreur quadratique moyenne (MSE) et le coefficient de corrélation (R).

Les tableaux I, II, III et IV représentent les meilleures performances trouvées pour les différentes combinaisons de couples de transfert pour chacun des 4 algorithmes, les figures 4, 5, 6, 7 représentent la variation de l'erreur quadratique moyenne (MSE) en fonction des couples de fonctions de transfert pour les 4 algorithmes.

D'après ces tableaux nous notons que:

- Avec le couple de fonctions de transfert (Logsig - Purelin) et une architecture [7-7-1] nous arrivons aux meilleures performances pour l'algorithme (Resilient Back Propagation).
- Pour l'algorithme (Gradient Descent Back Propagation), la meilleure performance est obtenue avec un réseau de neurones d'architecture [7-4-1], avec la fonction Purelin comme fonctions de transfert pour la couche cachée et la fonction Logsig pour la couche de sortie.
- Nous avons obtenu les meilleurs résultats pour l'algorithme (Gradient Conjugué Scalaire) avec un réseau de neurones d'architecture [7-8-1] et qui a comme couple de fonctions d'activations (Logsig-Tansig).
- L'architecture [7-5-1] avec une fonction Tansig pour la couche d'entrée et une fonction Purelin pour la couche de sortie, a donné les meilleures performances pour l'algorithme Levenberg-Marquardt.
- Comme la combinaison (Purelin-Purelin) n'a pas abouti aux meilleures performances, cela signifie que la relation entre l'humidité relative et les autres paramètres météorologiques est non linéaire.

✓ **Au niveau du nombre d'itérations :**

- L'algorithme RBP est l'algorithme présentant le plus grand nombre d'itérations pour toutes les combinaisons de fonctions de transfert ; cet algorithme est réputé très performant dans la reconnaissance des formes que dans l'approximation des fonctions.
- L'algorithme LM converge avec le minimum du nombre d'itérations pour toutes les combinaisons de fonctions de transfert ; cet algorithme est réputé très performant dans l'approximation des fonctions surtout quand le réseau contient moins d'une centaine de poids à mettre à jour, qui est le cas ici [13, 14].
- Le nombre d'itérations des différentes variantes de l'algorithme gradient conjugué scalaire est généralement inférieur ou égal à cent à l'opposé de l'algorithme RBP. Cependant, tous ces algorithmes convergent généralement avec un nombre d'itérations plus grand que celui de l'algorithme LM. Ces algorithmes sont connus dans la littérature par leur convergence rapide sur des variétés des problèmes (approximation des fonctions, classification), surtout quand la taille du réseau est assez grande.

Malgré ces différences, tous ces algorithmes sont de hautes performances, ils convergent rapidement, et aboutissent à des faibles valeurs de l'erreur quadratique moyenne MSE et de grandes valeurs du coefficient de corrélation R et cela dans un temps qui ne dépasse pas quelques secondes.

Tableau I: Etude de l'algorithme Resilient Back Propagation : RBP

Couche cachée	Couche de sortie	Désignations	R	MSE (10^{-5})	Architecture	Nombre d'itérations
Tansig	Tansig	TT	0,951	13,5	[7-4-1]	140
Tansig	Logsig	TL	0,911	29,5	[7-6-1]	160
Tansig	Purelin	TP	0,910	29,6	[7-4-1]	150
Logsig	Logsig	LL	0,911	28,8	[7-5-1]	120
Logsig	Tansig	LT	0,910	25,1	[7-6-1]	70
Logsig	Purelin	LP	0,970	06,4	[7-7-1]	100
Purelin	Purelin	PP	0,937	17,4	[7-8-1]	120
Purelin	Logsig	PL	0,919	29,4	[7-8-1]	90
Purelin	Tansig	PT	0,928	20,1	[7-6-1]	50

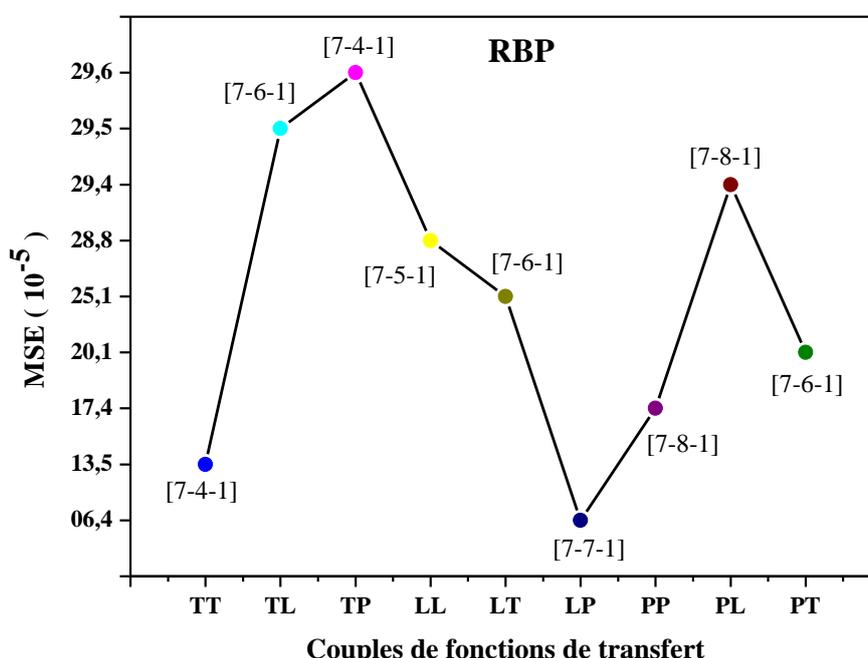


Fig.4 : Variation du MSE en fonction des couples de transfert pour l'algorithme RBP.

Tableau II : Etude de l'algorithme Gradient Descent Backpropagation GD

Couche cachée	Couche de sortie	Désignations	R	MSE (10^{-5})	Architecture	Nombre d'itérations
Tansig	Tansig	TT	0,832	85,3	[7-5-1]	110
Tansig	Logsig	TL	0,793	68,6	[7-7-1]	80
Tansig	Purelin	TP	0,847	42,0	[7-6-1]	70
Logsig	Logsig	LL	0,675	80,5	[7-5-1]	40
Logsig	Tansig	LT	0,719	99,8	[7-7-1]	60
Logsig	Purelin	LP	0,800	69,4	[7-5-1]	50
Purelin	Purelin	PP	0,720	99,3	[7-5-1]	45
Purelin	Logsig	PL	0,924	31,2	[7-4-1]	60
Purelin	Tansig	PT	0,804	66,1	[7-6-1]	50

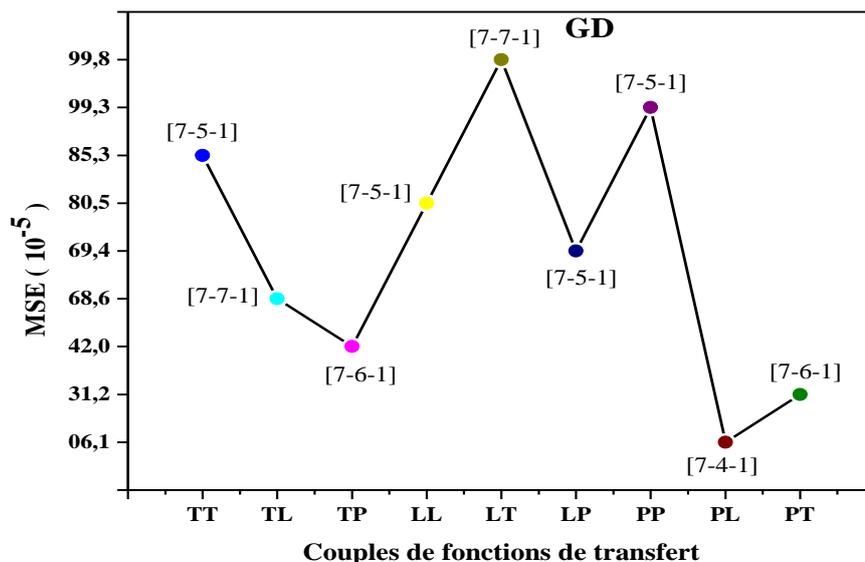


Fig.5: Variation du MSE en fonction des couples de transfert pour l'algorithme GD

Tableau III : Etude de l'algorithme Gradient conjugué scalaire (GCS)

Couche cachée	Couche de sortie	Désignations	R	MSE (10^{-5})	Architecture	Nombre d'itérations
Tansig	Tansig	TT	0,956	13,6	[7-5-1]	90
Tansig	Logsig	TL	0,918	28,0	[7-4-1]	80
Tansig	Purelin	TP	0,958	11,5	[7-6-1]	70
Logsig	Logsig	LL	0,921	27,8	[7-7-1]	100
Logsig	Tansig	LT	0,960	11,0	[7-8-1]	100
Logsig	Purelin	LP	0,955	12,6	[7-5-1]	80
Purelin	Purelin	PP	0,937	17,3	[7-4-1]	40
Purelin	Logsig	PL	0,917	28,0	[7-5-1]	50
Purelin	Tansig	PT	0,939	17,0	[7-6-1]	70

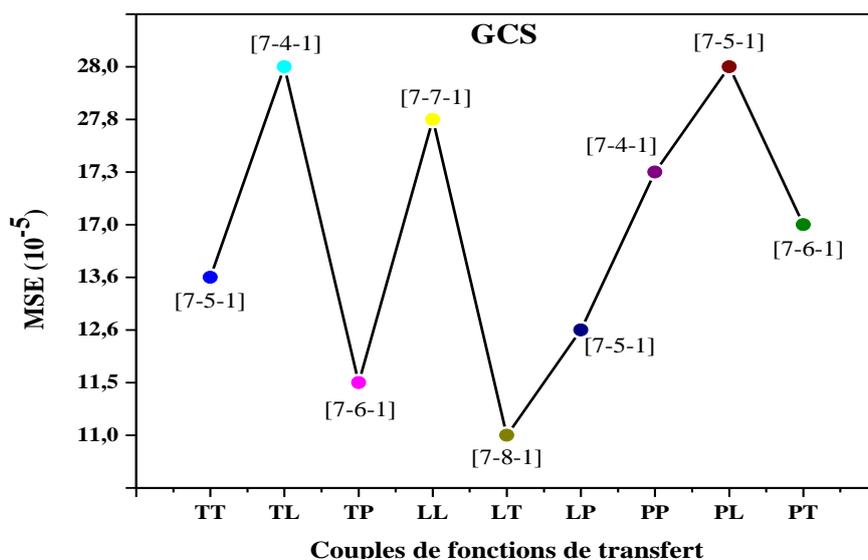


Fig.6 : Variation du MSE en fonction des couples de transfert pour l'algorithme GCS

Tableau IV: Etude de l'algorithme Levenberg-Marquardt (LM)

Couche cachée	Couche de sortie	Désignations	R	MSE (10^{-5})	Architecture	Nombre d'itérations
Tansig	Tansig	TT	0,960	8,50	[7-8-1]	50
Tansig	Logsig	TL	0,922	27,0	[7-9-1]	30
Tansig	Purelin	TP	0,970	1,50	[7-5-1]	27
Logsig	Logsig	LL	0,927	25,80	[7-10-1]	60
Logsig	Tansig	LT	0,965	7,50	[7-9-1]	30
Logsig	Purelin	LP	0,963	7,80	[7-11-1]	30
Purelin	Purelin	PP	0,930	17,20	[7-10-1]	30
Purelin	Logsig	PL	0,918	27,90	[7-4-1]	60
Purelin	Tansig	PT	0,939	17,00	[7-6-1]	90

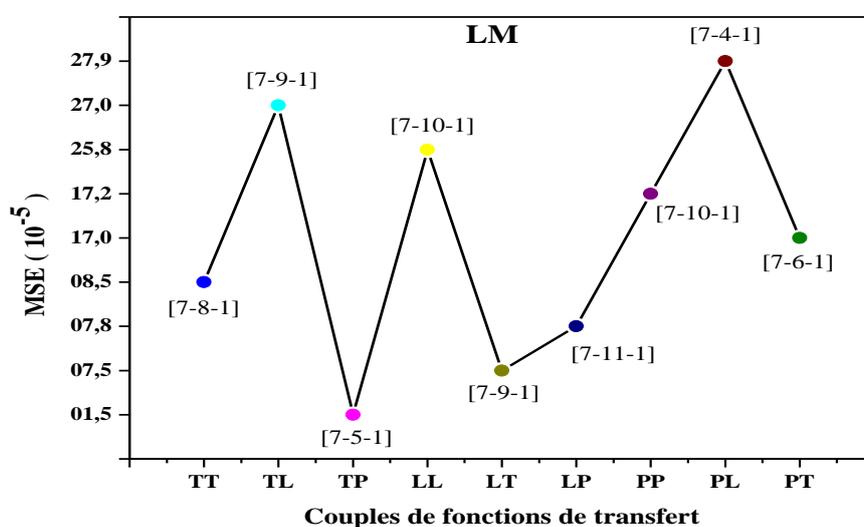


Fig.7 : Variation du MSE en fonction des couples de transfert pour l'algorithme LM

Le tableau V donne les couples de fonctions de transfert selon leurs performances pour les différents algorithmes d'apprentissage.

D'après ce tableau, on peut diviser les couples de fonctions de transfert en trois groupes :

- Le premier groupe donne des résultats plus performants que ceux relatifs aux autres couples de fonction de transfert.
- Le deuxième groupe donne des performances situées entre le premier groupe et le troisième, mais qui sont plus proches de celles de premier groupe.
- Le troisième groupe est constitué des couples avec des couples de fonctions qui présentent de faibles performances que celles relatives aux autres couples de fonctions de transfert.

Tableau V : Couples de fonctions de transfert selon leurs performances.

	<i>Groupe 1</i>	<i>Groupe 2</i>	<i>Groupe 3</i>
	<i>Meilleures performances</i>	<i>Moyennes performances</i>	<i>Faibles performances</i>
<i>RBP</i>	TP, TL	LT, PP, PL, LL, PT	TT, LP
<i>GD</i>	LT, TP	TL, LL, LP, TP, TT	LP, PT
<i>GCS</i>	TL, LL, PL	LL, LP, PP, PT	LT, TP
<i>LM</i>	TL, LL, PL	PT, LL, LP, PP, PT	TP

C. Choix du modèle RNA

Par la suite, nous allons comparer ces différents algorithmes afin de choisir le meilleur modèle. Le tableau VI, représente les meilleures performances obtenues pour chaque algorithme. Parmi tous ces algorithmes, l'algorithme de Levenberg–Marquardt est l'algorithme le plus performant au niveau des indicateurs statistiques (R et MSE), ainsi qu'au niveau de la rapidité de convergence, car il mélange les algorithmes de Gradient Descent et de Gauss-Newton. Pour des petites valeurs du pas, la méthode de Levenberg–Marquardt s'approche de celle de Newton. Inversement, pour de grandes valeurs du pas. Ceci est en accord avec les résultats de quelques travaux récentes qui ont démontré les capacités de l'algorithme d'apprentissage de Levenberg Marquardt pour la prédiction en terme d'erreur de prédiction ainsi le temps d'apprentissage d'une part et d'autre part cet algorithme d'apprentissage est un standard pour l'optimisation de l'erreur quadratique due à ses propriétés de convergence rapide et à sa test de robustesse [15, 16, 17].

De plus, après plusieurs tests, il est aisé de conclure que c'est celle qui allie le mieux, pour notre travail, rapidité et qualité. Son principe est basé sur une descente de gradient itérative de la famille quasi-Newton. Elle s'appuie sur les techniques des moindres carrés non-linéaires et de l'algorithme de Gauss-Newton à voisinage restreint, conférant une garantie rapide de la convergence vers la valeur minimal de l'erreur quadratique moyenne. Le pas de taux d'apprentissage η est initialisé avec une certaine valeur qui est modifiée durant l'optimisation. A chaque itération, la fonction de cout est calculée avec la valeur de η précédente, si la fonction de cout diminue, on effectue la modification des paramètres et on diminue η , si la fonction de cout croit, nous cherchons à se rapprocher du gradient et nous augmentons η jusqu'au ce que la fonction de cout diminue. Dans le cadre de ce travail, l'algorithme LM a été initialisé avec les valeurs des paramètres d'apprentissage suivantes :

$$\eta = 1 ; \text{Gradient} = 0,1 ; \text{Nombre maximale d'itérations} = 200$$

L'algorithme LM a convergé après 7 itérations dans un temps de 2s et ses paramètres sont devenus :

- $\eta = 0,001$
- Gradient = 0,0026

Tableau VI : Meilleures performances obtenues pour chaque algorithme d'apprentissage.

<i>Algorithmes d'apprentissage</i>	<i>Couche cachée</i>	<i>Couche de sortie</i>	<i>Désignations</i>	<i>R</i>	<i>MSE (10^{-5})</i>	<i>Architecture</i>	<i>Nombre d'itérations</i>
RBP	Logsig	Purelin	TT	0,969	06,4	[7-7-1]	100
GD	Purelin	Tansig	TP	0,924	31,2	[7-6-1]	60
GCS	Logsig	Tansig	LP	0,960	11,0	[7-8-1]	140
LM	Tansig	purelin	TP	0,970	1,5	[7-5-1]	27

Nous pouvons conclure d'après tous ces résultats, que le meilleur modèle RNA est un réseau de neurones de type PMC qui utilise l'algorithme de Levenberg Marquardt comme algorithme d'apprentissage, les fonctions (Tansig et Purelin) comme fonctions de transfert et qui a l'architecture [7-5-1]. Ce réseau est constitué de trois couches:

- Une couche d'entrée : formée par 7 neurones représentant les variables explicatives: la température de l'air, la température de rosée, la pression atmosphérique, la visibilité, la nébulosité, la vitesse du vent et la pluie. Dans cette couche aucun calcul ne se fait.
- Une couche cachée, ou se font tous les calculs d'optimisation des paramètres des réseaux de neurones.
- Une couche de sortie du réseau qui représente l'humidité relative.

La figure 8 représente l'architecture du modèle des réseaux de neurones développé :

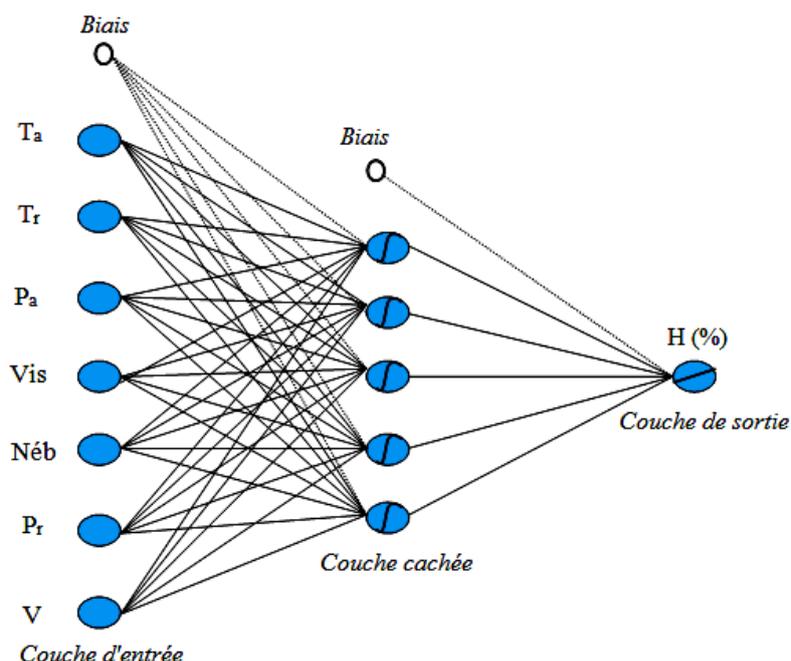


Fig.8: Architecture du réseau de neurones à trois couches de configuration [7, 5,1].

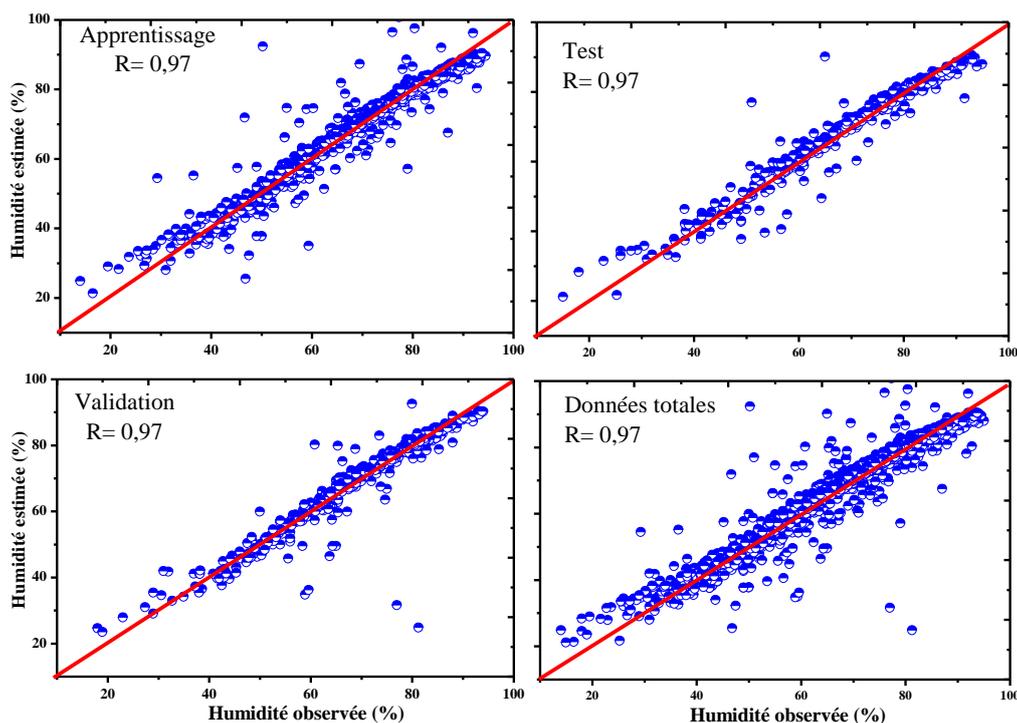


Fig.9: Relation entre les valeurs observées et celles estimées par le modèle RNA pendant les différentes phases (Apprentissage, validation, test ainsi que pour l'ensemble de la base de données)

Le coefficient de corrélation obtenu par ce modèle est $R=0,97$ et l'erreur quadratique moyenne est $MSE=1,5.10^{-5}$

Le modèle développé a donné le même coefficient de corrélation (0,97) durant les trois phases d'apprentissage, de test et de validation (fig.9).

D'après ces résultats numériques, nous pouvons conclure que les modèles neuronaux ont des coefficients de corrélation presque égal à l'unité ($R \sim 1$). Ceci montre qu'il y a un grand rapprochement entre les valeurs mesurées et celles estimées par le modèle neuronal développé dans cette étude. Ce modèle et peut donc être considéré comme un outil d'une grande efficacité dans le cadre de la prévision météorologiques. Nous citons a

titre d'exemples quelques travaux de recherche pertinents qui montrent l'efficacité de ce modèle dans la prévision météorologiques :

Bouaoune et al., 2010, [18] ont présenté l'application d'une méthode d'interpolation des réseaux neuronaux. Ils ont montré des relations non linéaires entre les variables météorologiques.

Perez et al., 2001, [19] ont proposé de prévoir la concentration de NO_2^- et de l'oxyde nitrique NO^- à Santiago en se basant sur les variables météorologiques. Leurs résultats ont montré que le réseau de neurones est la méthode qui réalise l'erreur de prévision la moins élevée par rapport aux autres méthodes classiques.

Cheggaga et al. 2010, [20] ont montré la possibilité ainsi l'efficacité de l'utilisation des réseaux de neurones à couches non-récurrentes pour la prédiction de la vitesse de vent à partir des données météorologiques.

D. Comparaison entre les deux modèles

Cette comparaison sera effectuée au niveau des indicateurs statistiques R et MSE. D'après le tableau VI, les meilleures estimations sont obtenues à l'aide du modèle RNA, car elle présente les plus faibles des critères statistiques [21].

Le modèle RNA garde les mêmes performances pendant les trois phases d'apprentissage, de validation et de test à l'opposé du modèle RLM qui montre une dégradation de performances en passant de la phase d'apprentissage au deux autres phases. Les deux modèle présentent une performance similaire durant la période d'apprentissage avec un coefficient de corrélation $R=0,97$ pour le modèle RNA et $R=0,86$ pour le modèle RLM.

Le modèle RNA permet des améliorations jusqu'à 13% de l'explication de la variance pour la phase de validation et la phase de test par rapport au modèle RLM.

Tableau VII: Comparaison des performances de deux modèles (RNA et RLM) au niveau des indicateurs statistiques.

<i>Phase</i>	<i>Apprentissage</i>		<i>Validation</i>		<i>Test</i>	
<i>Indicateur statistique</i>	<i>R</i>	<i>MSE</i>	<i>R</i>	<i>MSE</i>	<i>R</i>	<i>MSE</i>
RLM	0,86	0,150 0	0,84	0,7700	0,84	0,7700
RNA	0,97	0,001 5	0,97	0,0015	0,97	0,0015

V. CONCLUSIONS

Les réseaux de neurones artificiels sont des outils très puissants au niveau de la prédiction. Ils peuvent traiter des problèmes non linéaires. Mais ils présentent un inconvénient majeur du choix de l'architecture du réseau, car ce choix appartient souvent à l'utilisateur. Dans notre étude, nous avons développé plusieurs modèles basés sur des algorithmes d'apprentissage qualifiés de haute performance. Les résultats obtenus démontrent que l'algorithme de Levenberg-Marquardt présenté les meilleures performances aussi bien qu'au niveau des indicateurs statistiques qu'au niveau de la rapidité de convergence, ainsi qu'un réseau d'architecture [7-5-1] avec une fonction d'activation non linéaire de type Tansig pour la couche cachée et une fonction d'activation linéaire de type Purelin dans la couche de sortie. Ce réseau neuronal élaboré a permis une très bonne prédiction du taux d'humidité dans la région de Chefchaouen pendant les cinq années de la période d'étude. L'étude comparative entre les deux modèles RNA et RLM a montré que le modèle développé à base des réseaux de neurones artificiels est plus performant que celui établi par la méthode de régression linéaire multiple. Cela montre que la relation reliant l'humidité relative aux autres paramètres météorologiques dans la zone de Chefchaouen est non linéaire.

REFERENCES

- [1]. S. Abdelli, K. Nouria, A. Trabelsi, « Prévision du NO_2 en utilisant la méthode du réseau de neurones », Business and Economic Statistics Modeling Laboratory Institut Supérieur de Gestion de Tunis, 129-140p, 2012.
- [2]. E. Fock, P. Lauret, T. Mara et H. Boyer, « Modélisation hybride pour la simulation thermique du bâtiment », Congrès français de Thermique, SFT 2000, Lyon, 15-17 mai, 2000.
- [3]. A. Abdallaoui, « Le phosphore et les métaux lourds contenus dans les sédiments. Leur influence sur les phénomènes d'eutrophisation et de pollution », Editions Universitaires Européennes, 348 p. 2013.

- [4]. H. El Badaoui, A. Abdallaoui, I. Manssouri et H. Ousmana, "The prediction of moisture through the use of neural networks MLP type , "IOSR Journal of Computer Engineering Vol. 11, 66-74p, 2013.
- [5]. H. Vassiliki Mantzari, H. Dimitrios Mantzaris, "Solar radiation: Cloudiness forecasting using a soft computing approach, " Journal Artificial Intelligence Research, Vol 2 n° 1, p. 69-80, 2013.
- [6]. M. A. Ammar, " Mise en oeuvre de réseaux de neurones pour la modélisation de cinétiques réactionnelles en vue de la transposition batch/continu, " Thèse de doctorat de l'I.N.P. – Toulouse –194 p, 2007.
- [7]. S. Chabaa, " Identification des systèmes non linéairement utilisant les techniques d'intelligence artificielles et les bases de fonctions de la guerre pour la modélisation des données du trafic dans les réseaux internet, " Thèse de Doctorat, Université Cadi Ayyad, Faculté des Sciences Semlalia-Marrakech, 187 p, 2011.
- [8]. I. Manssouri, M. Manssouri, B. El Kihel, "Fault Detection by K-NN algorithm and MLP neuronal networks in distillation column," Journal of information, Intelligence and knowledge, Vol.3, pp.57-72 2011.
- [9]. M. Bélanger, N. El-Jabi, D. Caissie, F. Ashkar, J. M. Ribí, " Estimation de la température de l'eau en rivière en utilisant les réseaux de neurones et la régression linéaire multiple, " Revue des sciences de l'eau, 18,3: 403-421, 2005.
- [10]. G. Mustafaraja, "Prediction of room temperature and relative humidity by autoregressive linear and nonlinear neural network models for an open office," Energy and Buildings, p. 1452–1460, 2011.
- [11]. A. Zouidi, A. Chaari, M. Stambouli et Fnaiech F., " Nonlinear continuous time modeling of a high pressure mercury vapor discharge lamp using feed forward back-propagation neural networks, " IEEE 2004, Yasmine Hammamet, Tunisie, 2004.
- [12]. K. Dahmani, G. Notton, R. Dizène et C. Paoli, "Etat de l'art sur les réseaux de neurones artificiels appliqués à l'estimation du rayonnement solaire, "Revue des Energies Renouvelables, Vol. 15 N°4 687 – 702, 2012.
- [13]. O. Kisi, E. Uncuoglu, "Multi-layer perceptrons with Levenberg-Marquardt training algorithm for suspended sediment concentration prediction and estimation, " Journal des Sciences Hydrologiques, 16p, 2004.
- [14]. O. Kisi, E. Uncuoglu, "Comparison of three back-propagation training algorithms for two cases studies," Indian Journal of Engineering and Materials Sciences, Vol.12, October, p.434-442, 2005.
- [15]. L. Hontoria, J. Aguilera, Riesco, J. Zufiria, "Recurrent neural supervised models for generating solar radiation," Journal of Intelligent and Robotic Systems, Vol. 31, 201–221p, 2001.
- [16]. A. Mellit, S.A. Kalogirou, L. Hontoria, S. Shaari, "Artificial intelligence techniques for sizing photovoltaic systems, "A review, Renewable and Sustainable Energy Reviews, Vol 13, 406-419 p, 2009.
- [17]. C. Voyant, "Prédiction de séries temporelles de rayonnement solaire global et de production d'énergie photovoltaïque à partir de réseaux de neurones artificiels, " Thèse de doctorat, Université de Corse-Pascal Paoli, Ecole Doctorale Environnement et Société / UMR CNRS 6134 (SPE), 257p, 2011.
- [18]. D. Bouaoune, M. Dahmani, "Reconstitution de données climatiques pour l'Algérie du Nord : application des réseaux neuronaux, "Comptes Rendus Géoscience 342, 815–822, 2010.
- [19]. P. Perez, A. Trier, "Prediction of NO and NO2 concentrations near a street with heavy traffic in Santiago », Atmospheric Environment,"Vol° 35, 1783-1789p, 2001.
- [20]. N. Cheggaga, F. Youcef Ettoumi, "Estimation du potentiel éolien, "Revue des Energies Renouvelables, 99 – 105p, 2010.
- [21]. N. Sarani, J. Soltani, S. Sarani, A. Moasheri, "Comparison of Artificial Neural Network and Multivariate Linear Regression Model to Predict Sodium adsorption ratio (SAR) (Case Study: Sistan River, Iran) , " International Journal of Biological, Ecological and Environmental Sciences, Vol°1, 130-134p, 2012.