International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 02 Issue: 08 | Nov-2015 www.irjet.net p-ISSN: 2395-0072

Prediction of air temperature using Multi-layer perceptrons with Levenberg-Marquardt training algorithm

Mustapha BEN EL HOUARI[1, 2], Omar ZEGAOUI[1], Abdelaziz ABDALLAOUI[2*]

1) *University Moulay Ismail, Faculty of Sciences, Department of Chemistry, Research Team "Matéraiux et Catalyse Appliqués", BP 11201, Zitoune, Meknes, Morocco.*
2) *University Moulay Ismail, Faculty of Sciences, Department of Chemistry, Research Team "Analytical Chemistry and Environment", BP 11201, Zitoune, Meknes, Morocco.*

Abstract - ***In this study, we developed neural models related to the prediction of the air temperature in the region of Meknes in Morocco. Depending on weather parameters, such as atmospheric pressure (Pr), humidity (H), visibility (Vis), wind speed (V), dew point (Tr) and the precipitation (P). The database used contains a history of meteorological parameters covering 3288 days, from 2004 to 2012. This multilayer perceptron type of neuron (MLP) was used to approximate the relationship between these parameters with minimum square error. For a better air temperature prediction, we developed a stochastic neural model. The mean square error (MSE) and the correlation coefficient (R) were used to evaluate the performance of the developed models. The study of these statistical indicators demonstrated that the prediction of the air temperature was powerful with the Levenberg-Marquard algorithm, with architecture [6-10-1], the Tansig function in the hidden layer and the Purelin function in the output layer.***

Key Words: *Key ANN, MLP, Levenberg-Marquard algorithm, MSE, R, Air temperature.*

1. INTRODUCTION

The effects of climate change, including droughts, floods, tornadoes, severe weather, rising sea levels, and industrial activities may result in food shortages, an increase in vector-borne diseases, infrastructure damage, and deterioration of natural resources from which people derive their livelihoods. Most human activities are directly related to weather.

Indeed, knowing the variability of ambient temperature is important in agriculture because extreme changes in air temperature may cause damage to plants and animals **[1;2]**. Air temperature forecasting is useful in knowing the probability of tornadoes and flood occurrence in a specific area **[3]**. Prediction of the energy consumption, soil surface temperature, and solar-radiation are also related to ambient air temperature forecasting **[1]**.

Some studies have applied different models in forecasting such as Artificial Neural Network. Smith *et al*. **[2]** founded that extreme cold and heat can affect crops and livestock. They predicted ambient air temperature using artificial neural networks with a time step of 1 to 12 hours. Their results showed that neural networks can predict air temperature throughout the year with a minimum mean absolute error (MAE).

Other authors **[3]** have used artificial neural networks to predict air temperature in Jeddah in Saudi Arabia. They founded effective results even with a single entry. Also, Dombayc *et al.* **[1]** have used artificial neural networks to predict the daily ambient temperature in Denizli in southwest Turkey. These authors separated the meteorological data into two parts while choosing the period from 2003 to 2005 for training data and 2006 for testing data. They developed a model of retro-propagation of error that has three entrances, six hidden neurons and one output.

Behrang *et al.* **[4]** developed the multilayer Perceptron (MLP) and radial basis function (RBF) neural networks for the prediction of daily sunlight. They considered different weather variables including average air temperature, relative humidity, sunshine hours, evaporation and values of the wind speed between 2002 and 2006 for the city of Dezful in Iran. The results obtained by these authors showed the robustness of the artificial neural networks. El Badawi *et al*. **[5]** have developed statistical models of neuronal type Multilayer Perceptron (MLP) and Network Radial Basis Function (RBF) for the prediction of humidity. They treated a time series of meteorological data and showed the robustness of the artificial neural network Multilayer Perceptron type.

Other studies **[6;7;8;9;10]** have confirmed that the use of artificial neural networks is effective to predict weather parameters, especially the air temperature with high accuracy. Artificial neural networks are sophisticated techniques data processing, able to model the relationship between particularly complex functions. Over the past two decades, neural networks have proven themselves in many areas. Several studies have confirmed that they are also well suited for the prediction of meteorological parameters **[1;11]**.

The objective of this work is to develop a multi-layer perceptron model type (MLP) for the prediction of air temperature in the region of Meknes in Morocco according to other meteorological parameters by performing the study of the effect of different data distributions, learning algorithms, transfer functions, the number of hidden layers, hidden neurons, and the performance of the established mathematical models.

2. MATERIAL AND METHODS

2.1 Database

The database used for predicting the air temperature contains a history of seven meteorological parameters covering 3288 days (from 2004 to 2012). Table 1 presents these variables and their designations.

The database used for predicting the air temperature contains a history of seven meteorological parameters covering 3288 days (from 2004 to 2012). Table 1 presents these variables and their designations.

Table -1: Meteorological variables and their designations

Variable	Designation	Unit	variable
Air temperature	T	°C	Dependant
Atmospheric pressure	Pr	hpa	Independant
Humidity	H	%	Independant
Visibility	V	km	Independant
Wind speed	Vvent	km/h	Independant
Dew point	Tr	°C	Independant
Precipitation	P	mm	Independant

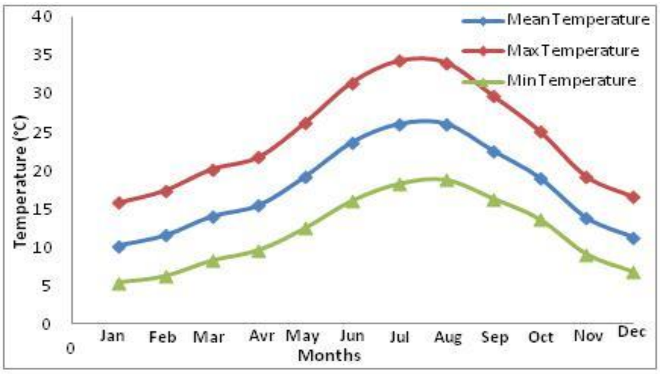


Chart-1: Mean, maximum and minimum average air temperature for each month during the period studied

Chart -1 shows the evolution of mean, max and mean average air temperature for each month during the period studied. The evolution of air temperature can be separated in three domains:

- ✓ January-April: the monthly increase in air temperature is gradual and steady. It is of the order of 1 to 2°C per month;
- ✓ May-August: a monthly increase is of the order of 3 to 5°C per month;
- ✓ September-December: a reduction of the order of 2 to 5°C per month.

2.2 Data Normalization

In general, the database has to undergo a pre-treatment to be tailored to the network inputs. The learning base consists of six vectors X_1, X_2, ..., X_6. These data are converted into standardized variables. Indeed, the values of each independent variable Xi were standardized relative to its mean value and its standard deviation according to equation (1):

$$X_{ijs} = \frac{X_{ij} - X_{im}}{\sigma_{ix}} \qquad (1)$$

X_{ijs}: standardized value of variable Xi for observation j,

X_{ij} : value of variable Xi for observation j,

X_{im}: average value,

σ_{ix}: the standard deviation of the variable X_i.

Similarly, values of the dependent parameter are normalized in the range [0, 1] from the equation (2):

$$Y_{jn} = \frac{Y_j - Y_{min}}{Y_{max} - Y_{min}} \qquad (2)$$

Y_{jn}: normalized value of the variable Y for observation j,

Y_j: actual value of the variable Y for observation j,

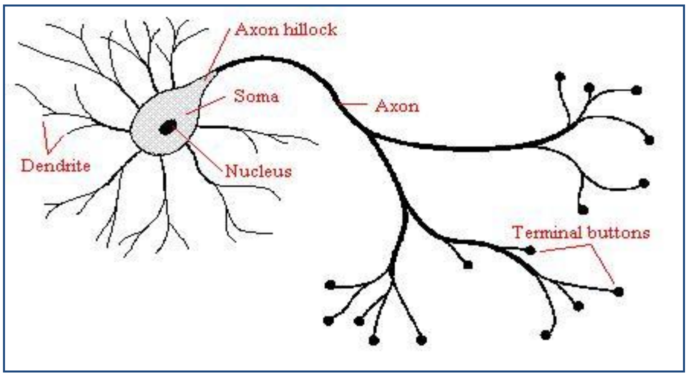
Y_{max}: maximum value of the variable Y,

Y_{min}: minimum value of the variable Y.

2.2 General Operation Of Neural Networks

- **The formal neuron**

The technique of neural networks is based on the operation of the nerve cell. The biological neuron consists of a cell body which receives information via its dendrites, and if the received activity exceeds a certain threshold, it sends a response via its axon (Figure 1).



International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 02 Issue: 08 | Nov-2015 www.irjet.net p-ISSN: 2395-0072

Fig -1: Drawing of biological neuron

The formal neuron gets the simplified operation of the biological neuron **[12]**. Each neuron has multiple inputs denoted by Xi, from which it calculates an output denoted by S **[13]**.

More concretely, each input is weighted by a synaptic weight denoted by W_i, and an activation function. It then performs a weighted sum of these inputs given as follows:

$$[\sum_{i=1}^{n} W_i \cdot X_i]$$

A transfer function φ, non-linear (sigmoid function, for example), and finally calculates the output S as a function of the value of the activation function, namely:

$$S = \varphi\, [\textstyle\sum_{i=1}^{n} W_i \cdot X_i] \qquad (3)$$

- **Artificial neural networks**

Artificial neural networks are characterized by a very particular structure. Its neurons are organized in successive layers where information flows in one direction, from the input layer to the output layer. Neurons of the same layer are not interconnected. A neuron can send the result to a neuron located in a back layer. The network is formed of several layers called by convention:

✓ Input layer: This layer is always a layer associated with the system inputs, consisting of input cells corresponding to n input variables;

✓ Layer (s) hidden (s): each layer is composed of m neurons, whose activation function is of the family of bounded curves "S". These neurons have no connection with the outside and are called hidden neurons;

✓ Output layer: it is constituted by the output values of the linear combination of the functions of the hidden layer **[14]**.

Neurons are connected together by weighted connections. They are the weights of these connections which govern the operation of the network and program an application of the input space to the output space, using the non-linear transformation. The main applications of neural networks are pattern recognition, classification, time series prediction, and modeling **[15]**. The structure of the three-layered artificial neural network is shown in Figure 2.

The transfer functions are sigmoid function, logsig function and linear function depicted respectively in Figure 3. The formula of tansig, logsig and purelin transfer functions are expressed respectively in equations 4, 5 and 6:

$$tansig\ (n) = \frac{1}{(1+\exp(-n))} \qquad (4)$$

$$logsig\ (n) = \frac{2}{(1+\exp(-2n))} - 1 \qquad (5)$$

$$purelin\ (n) = n \qquad (6)$$

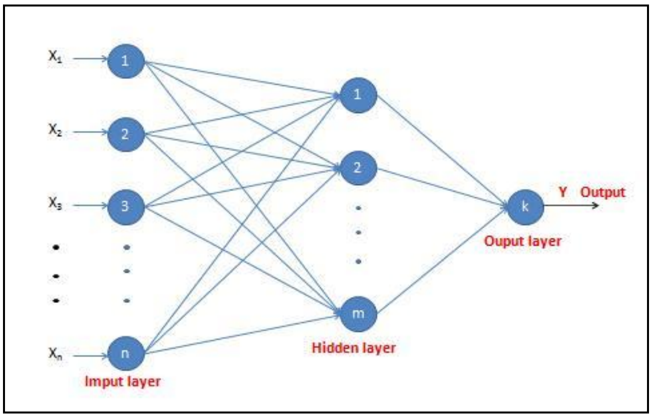


Fig -2: Diagram of the neural network of three-layer structure

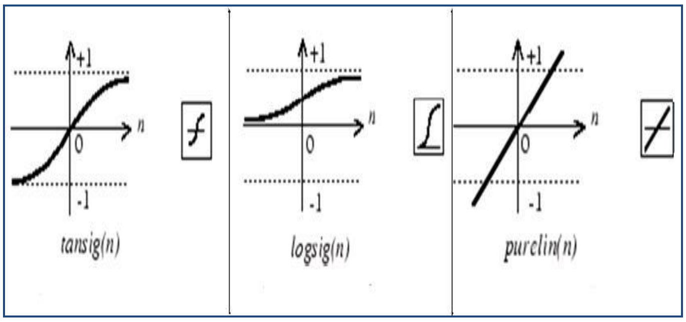


Fig -3: Schematic presentation of tansig, logsig and purelin transfer functions **[16]**

- **Performance evaluation**

The performance of the model efficiency was measured by using two criteria:

✓ The Mean Square Error (MSE) defined as:

$$MSE = \frac{1}{N}\sum_{1}^{N}(Y_{Pj} - Y_{Oj})^2 \qquad (7)$$

With Y_{Pj} et Y_{Oj} are the predicted and observed values respectively; N is the number of observations.

✓ The correlation coefficient (R) between the desired and the estimated values for each neuron of the output layer defined as:

$$R = \left[1 - \frac{\sum_j (Y_{Oj} - Y_{Pj})^2}{\sum_j (Y_{Oj} - Y_m)^2}\right]^{\frac{1}{2}} \qquad (8)$$

With Y_m the average of the observed values.

In analyzing the different predictive models, the best answers for the model, MSE tends to zero and R close to one.

3. RESULTS AND DISCUSSION

3.1 Learning base and test database

In order to develop mathematical models based on artificial neural networks, database was divided in two phases: learning and testing phases.

The values of R and MSE obtained (see table 2) indicate that the best distribution of the learning phase is 70% of the global database.

Table -2: Name of the Table Mean square error and correlation coefficient of the learning phase for different distributions of the database.

Distribution of the database	MSE x10^{+3}	R
Learning phase: 90% Testing phase : 10%	3.83	0.9348
Learning phase: 80% Testing phase: 20%	3.76	0.9371
Learning phase: 70% Testing phase: 30%	**3.65**	**0.9407**
Learning phase: 60% Testing phase : 40%	3.89	0.9335

3.2 Number of hidden layers

Based on the MSE and R values obtained and presented in table 3, the model is well optimized with a one hidden layer corresponding to a maximum value of the correlation coefficient and the minimum of the root mean square.

Table -3: MSE and coefficient values of the learning phase obtained for different hidden

Number of hidden layers	MSE x 10^{+3}	R
1	**3.65**	**0.9407**
2	3.88	0.9314
3	3.93	0.9285
4	4.18	0.9209

3.3 Number of neurons in the hidden layer

An optimal ANN architecture may be considered as the one yielding the best performance in terms of error minimization.

According to the results by increasing the number of neuron in the hidden layer, the best structure obtained was with 10 neurons in the hidden neurons corresponding to the lowest MSE value and the highest R value as presented in table 4.

Table -4: Mean square error and correlation coefficient values of the learning phase for different neurons in the hidden layer

Number of neurons in the hidden layer	MSE. 10^3	R
1	4.53	0.9206
2	4.58	0.9174
3	4.28	0.9242
4	4.07	0.9299
5	3.98	0.9329
6	3.90	0.9339
7	3.72	0.9352
8	4.02	0.9319
9	3.92	0.9333
10	**3.65**	**0.9407**
11	3.67	0.9387
12	3.73	0.9374
13	3.74	0.9351
14	3.69	0.9371
15	3.70	0.9363
16	3.85	0.9342
17	3.68	0.9378
18	3.82	0.9315
19	3.73	0.9353
20	3.80	0.9358

3.4 Activation function

The different activation functions such as sigmoid linear and Gaussian **[17;18;19]** are used to ensure the interconnection between two neurons in different layer

To determine the best transfer functions of activation couple, in the hidden layer and the output layer, different combinations between Tansig, Logsig and Purelin activation functions have been studied. The obtained results based on the values of MSE and R (Table 5) showed that the best combination corresponding to the minimum value of MSE and the highest value of R is the sigmoid function (tansig) for the hidden layer and the linear function (Purelin) for the output layer.

The couple sigmoid-linear functions can be adapted in an appropriate manner. The degree of nonlinearity of the model depends on the complexity of the problem. The model consists of an arrangement of neurons into several layers in which information propagates in a unidirectional way from the input layer to the output layer.

Table -5: Mean square error and correlation coefficient of the learning phase for different transfer functions

Hidden Layer	Output Layer	MSE x 10^{+3}	R
Tansig	Tansig	3.66	0.9404
Tansig	Logsig	24.9	0.7698
Tansig	**Purelin**	**3.65**	**0.9407**
Logsig	Tansig	3.77	0.9353
Logsig	Logsig	25.2	0.7789
Logsig	Purelin	3.99	0.9343
Purelin	Tansig	4.93	0.9163
Purelin	Logsig	25.2	0.7202
Purelin	Purelin	5.27	0.9084

3.5 Optimum learning algorithm

Fourteen learning algorithms were tested to obtain the fastest appropriate algorithm for the training process. The algorithms studied are: BFGS quasi-Newton (Trainbfg), Bayesian regularization (Trainbr), Powell-Beale conjugate gradient (Traincgb), Fletcher-Powell gradient conjugate (Traincgf), Polak-Ribière gradient conjugate (Traincgp), gradient descent (Traingd), dynamic gradient descent (Traingdm), gradient descent with adaptive learning rate (Traingda), dynamic gradient descent and adaptive learning rate (Traingdx), Levenberg-Marquardt (Trainlm), an intersecting step (Trainoss), random training with incremental learning (Trainr), elastic (Trainrp), and batch training with weight and bias learning rules (Trainscg).

Table -6: Mean square error and correlation coefficient of the learning phase for different learning algorithms

Learning algorithm	MSE x 10^{+3}	R
Trainbfg	4.15	0.9285
Trainbr	3.77	0.9364
Traincgb	4.39	0.9238
Traincgf	4.62	0.9177
Traincgp	4.27	0.9271
Traingd	13.2	0.7864
Traingdm	11.2	0.7775
Traingda	9.81	0.8068
Traingdx	5.36	0.9067
Trainlm	**3.65**	**0.9407**
Trainoss	4.48	0.9245
Trainr	4.85	0.9158
Trainrp	4.47	0.9228
Trainscg	5.03	0.9112

The study of the effects of these algorithms on the performance of neural models showed that the Levenberg-Marquardt algorithm was the most efficient algorithm based on statistical indicators (MSE & R) as well as on the convergence speed. These results are in agreement with the results of some recent works that showed that the high capacity of the Levenberg Marquardt algorithm to predict error **[17;18]**.

3.6 Performance models developed

The multilayer perceptron design consists of three layers that are the input layer, the hidden layer and the output layer. The number of neurons of the input layer is determined by the size of the input vector. In this study, there are six neurons. The output layer contains a one neuron representing the air temperature and the hidden layer contains ten neurons. Figure 4 shows the architecture of the network developed in this work.

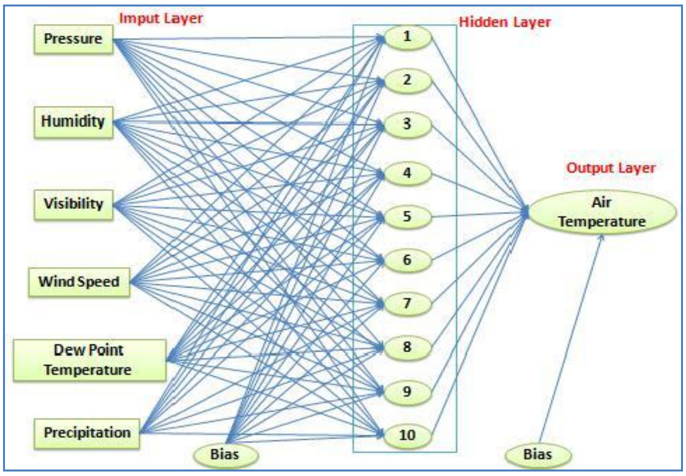


Fig -4: Architecture of MLP the neural network with three layers configuration [6-10-1] developed in this study

The neural network model developed in this study yielded a correlation coefficient of 0.9407 and a mean square error of 3.65 10^{-3}. The results shown in Chart 2 suggest a strong correlation between the observed results and those estimated from the air temperature during the studied period. The same conclusion was noted in the study of the residuals according the predicted values of air temperature. Chart-3 shows that the the residues fall on the abscissa axis zero.

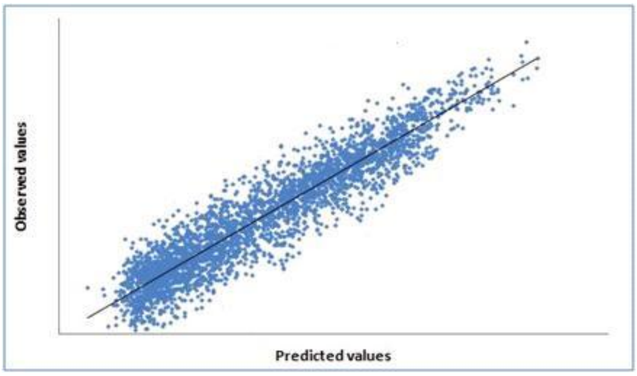


Chart-2: Relationship between the observed and estimated values of the air temperature.

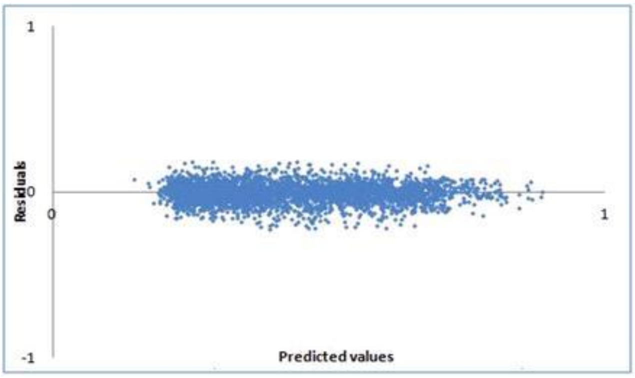


Chart-3: Distribution of residuals obtained by the developed MLP model for prediction of the air temperature

4. CONCLUSIONS

In this work, we have indicated the methods used in the development of artificial neural networks Multilayer Perceptron type to predict the air temperature in the region of Meknes in Morocco. We determined the distribution of the database, the learning algorithm of the transfer functions, the number of the hidden layer and the number of hidden neurons constituting the most optimal architecture that accelerates the convergence of the studied algorithms and minimize the mean square error.

The study of these statistical indicators demonstrates that the prediction of the air temperature is optimal and powerful with the Levenberg-Marquard algorithm having architecture [6-10-1]. The best transfer functions couple combination corresponds to the Tansig function in the hidden layer and the Purelin function in the output layer.

REFERENCES

[1] O. A. Dombayc and M. Golcu. Daily means ambient temperature prediction using artificial neural network method: A case study of Turkey. *Renewable Energy*, 34, 2009, 1158-1161.

[2] B. A. Smitha. G. Hoogenbooma. R. W. McClendon. Artificial neural networks for automated year-round temperature prediction. *Computers and Electronics in Agriculture*, 68, 2009, 52-61.

[3] I. Tasadduq. S. Rehman. K. Bubshait. Application of neural networks for the prediction of hourly mean surface temperatures in Saudi Arabia. *Renewable Energy*, 25, 2002, 545-554.

[4] M.A. Behrang. E. Assareh. A. Ghanbarzadeh. A. R. Noghrehabadi. The potential of different artificial neural network (ANN) techniques in daily global solar radiation modeling based on meteorological data. *Solar Energy*, 84, 2010, 1468-1480.

[5] H. El Badaoui, A. Abdallaoui, and S, Chabaa. Perceptron Multicouches et réseau à Fonction de Base Radiale pour la prédiction du taux d'humidité. *International Journal of Innovation and Scientific Research*, 5(1), 2014, 55-67.

[6] C. Devi, B. Reddy, K. Kumar, B. Reddy, N. Nayak. Artificial Neural Network Approach for Weather Prediction using Back Propagation, *International Journal of Engineering Trends and Technology*, 3, 2012, 19-23.

[7] QI. Moro, L. Alonso, CE. Vivaracho. Application of neural networks to weather forecasting with local data, *Proceedings of the 12th IASTED international conference on applied informatics Annecy France*, 1994 , 68-70.

[8] Maqsood, M.R. Khan and A. Abraham. An ensemble of neural networks for weather forecasting, *Neural Comput. & Applic*, 13, 2004, 113-122.

[9] A. Jain, R.W. McClendon, G. Hoogenboom, and R. Ramyaa. Prediction of frost for fruit protection using artificial neural networks, *American Society of Agricultural Engineers, St. Joseph, MI*, 2003, ASAE Paper 03-3075.

[10] J. C. M. Pires. F. G. Martins. Correction methods for statistical models in tropospheric ozone forecasting. *Atmospheric Environment*. 45, 2011, 2413-2417.

[11] K. Philippopoulos, D. Deligiorgi, and G. Kouroupetroglou. An artificial neural network approach for the forecast of ambient air temperature. *Geophysical Research Abstracts*, 16, 2014, 11446.

[12] S. Haykin. Neural Networks: A Comprehensive Foundation, *2nd edition. Prentice Hall PTR*. 1998.

[13] S. Haykin, Neural Neural Network and Its Application in IR, *a comprehensive foundation, 2nd edition Upper Saddle Rever, New Jersey: Prentice Hall*, 1999, 842.

[14] A. J. P, Kumar, and D.J.K, Singh. Artificial neural network-based wear loss prediction for a 390 aluminium alloy, *Journal. of Theoretical and Applied Information Technology*, 4(10), 2008, 961-964.

[15] S. Haykin. Les réseaux de neurones. une base globale. *Upper Saddle River, NJ: Prentice-Hall*, 1999.

[16] H. Demuth. M. Beale. M. Hagan. Matlab neural network toolbox user's guide, version 5.1. *The MathWorks Inc*, 2007.

[17] H. El Badaoui. Simulation numérique par l'utilisation des techniques d'intelligences artificielles pour la modélisation des données météorologiques. Thèse de Doctorat, Université Moulay Ismail, Faculté des Sciences Meknès, 210 p, 2014.

[18] H. El Badaoui, A. Abdallaoui, S. Chabaa. Etude des effets des algorithmes d'apprentissage et des fonctions de transfert sur la performance des modèles statistiques neuronaux: Application dans le domaine météorologique. *International Journal of Engineering Research and Development (IJERD)*, Volume 9, Issue 6, 2013, 48-56.

[19] K. Dahman, 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*, volume 15, 2012, 687–702.