Neuron Network Prediction Feed-Forwad Wind Speed Network on Mauritania's North Coast: Ballawack Case

Soukeyna Mohamed, Diene Ndiaye, Sidi Mohamed Mustapha, Abdel Kader Mahmoud

Abstract: The assessment of the suitability of a wind system depends largely on the prediction of the wind potential. Indeed, the variability and uncertainty inherent in renewable energy sources can have a significant impact on accurate and reliable prediction of the power produced. Wind sources are needed at different time stages and at different altitudes. Thus, putting in place tools for predicting these wind resources is essential for their effective integration in the frame of electricity generation. In this context, the paper of this study is to propose a short-term wind energy prediction method through the formation of historical wind velocity data based on neural networks. This assessment involves modelling wind speed using ANN through the feed-forwad network. So, ANN are at the basis of adaptive identification methods and intelligent command laws. In this sense, first, the process of forecasting wind energy involves the creation of a raw data base, which is then filtered by probabilistic neural network. More concretely, the contribution of the work can be given in the form of technical results. These results start with a proposal of the theoretical models, then it is given the approach method that is used, then it is proposed the design of the system and the whole is closed by a performance evaluation. As far as performance evaluation is concerned, it is presented in the form of the results of analysed simulations of the forecast model. In practical terms, it should be noted that the proposed model also provides a high degree of accuracy for the measured data. In the end, normalized average absolute errors were recorded between 4.7% and 4.9%. As, it was found a regression factor R (measures the correlation between output-Target) between 91% and 96% for the site of the northern Mauritanian coast. This is largely acceptable for similar calculations.

Keywords: Artificial Neural Networks (ANN), MATLAB, Mathematical model, Mauritanian north coast, Wind speed prediction, Wind power prediction
Abbreviations: ANN, Artificial Neural Networks; GA, Genetic Algorithm

Soukeyna Mohamed*, Assistant Professor, Department of physic Nouakchott, (Applied Research Laboratory for Renewable Energies (LRAER), Université de Nouakchott, Al-Aasriya (UNA), Mauritanie E-mail: soukwewe@gmail.com

Diene Ndiaye, Associate Professor, Department of physic, saint-louis, (Laboratory of Electronic, Computing, Telecommunication and Renewable Energies (LEITER), UGB, Saint-Louis, Senegal.), Senegal E-mail: Ndiaye-diene@yahoo.fr

Sidi Mohamed Mustapha, Assistant Professor, Department of physic Nouakchott, (Applied Research Laboratory for Renewable Energies (LRAER), Université de Nouakchott, Al-Aasriya (UNA), Mauritanie E-mail: sidi-mustapha@yahoo.fr

Abdel Kader Mahmoud, Assistant Professor, Department of physic Nouakchott, (Applied Research Laboratory for Renewable Energies (LRAER), Université de Nouakchott, Al-Aasriya (UNA), Mauritanie E-mail: nakader@yahoo.fr

I. INTRODUCTION

It is important to note that in the bibliography [1, 2, 3, 4], a multitude of prediction methods of varying utility are discussed. If, the first gives the predictions of natural disasters that fall within the framework of the security of people and their property (example earthquakes in the Indonesian region),

on the other hand the second article talks about machine learning algorithms and validates the results using Mean Absolute Error (MAE) and Mean Square Error (RMSE).

It is also possible to cite the example in [3] for the improvement of technical systems on the basis of the genetic algorithm (GA).Without forgetting, the article [4] whose author proposes an innovative approach to finding response keywords from a given corpus of news or data titles, realized with the use of Gated Recurrent Unit (GRU). Thus, in the present paper, the problem posed is that a large part of the sites of the Mauritanian coast are isolated from the national electricity grid and their majority are inhabited by sinners, without drinking water for their food needs, without means of conservation for their fishery products. These problems can find the solution through the wind application which has environmental protection benefits of these sites. In fact, wind speed forecasts can help to determine and optimize the location of weather observation station networks that are able to detect sites with the best wind potential [5] estimate the electricity production associated with daily operations as in [6]. With regard to the accuracy of forecast models, to be predicted, an example is proposed in [7] which is the high-quality model by NWP (Numerical Wearther Prediction), but has not been used for wind power.On the other hand, the model proposed in [8] is available every hour within the forecast horizon up to 12 hours or more, to take into account the influence of local speed and wind direction. It uses a data exploration approach with a model neighbouring k-plus. In the end, the approach chosen for this paper which v& guide our work will be summarized as follows: in part (2) it will be presented the site, as, it will be proposed in (3) the structures of neuron networks of type feed- forwad. Then, the results will be proposed in part (3) as a theoretical development of the neural networks feed-forwad. In addition, it will be given in these results the approach of the algorithm. Then, it will be proposed the system design by data preparation which is followed by an evaluation of the performance of the prediction method and in the end, a conclusion will be made to close this work.

Retrieval Number: 100.1/ijitee.K78530991120
DOI: 10.35940/ijitee.K7853.1091220
Journal Website: www.ijitee.org

369

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

A. Brief description of study area

The locality of BALLAWACK is located in the northern zone of the Mauritanian coast. It is located on the edge of

This community is populated by about 440 people with infrastructure such as a naval training centre. The activity of the populations, as has been said, is distinguished exclusively by the exploitation of the resources of the fishery.

II. MATERIALS AND METHODS

the Atlantic Ocean between the town of Nouakchott to the south and the town of Nouadhibou to the north (latitude 18.52° and longitude 16.07°).

Structure of the neuron network feed- forwad

For figures 1 and 2, the input units are arranged in different layers (Reference to the vertical row of neurons) and are interconnected by weighted connections (synapses). Thus, Figure.1, shows a feed-forwad network which presents a function of its inputs and Figure.2 proposes a network of multi-layer neurons with input units and a hidden layer.
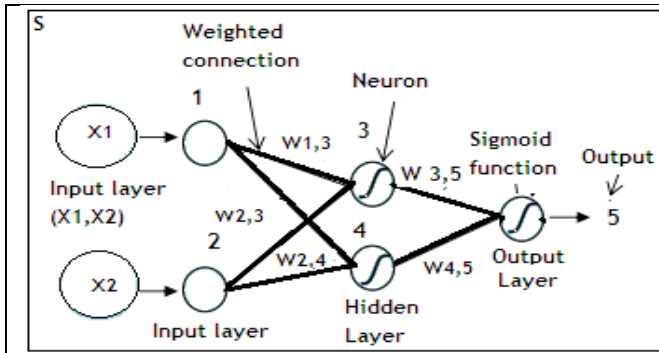
Fig. 1. Three-layered feed-forward NN architecture with neurons (or nodes) in the input layer (x1, x2), H = 2 neurons in the hidden layer (H) and no = 1 neuron (output layer).
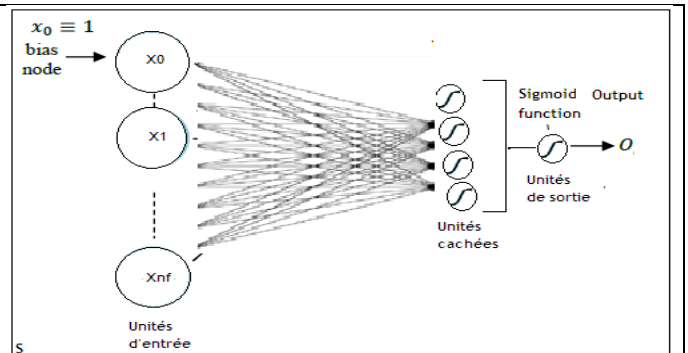
Fig.2. Network of multi-layer neurons with input units and a hidden layer.It is possible to propose this example in form (10 input units and 4 hidden units).

Theoretical development: It is assumed through the multi-layer network learning algorithm, an output vector $h_w(\mathbf{x})$ instead of a single value, and then for each example has a vector of outputs y. The major difference is that the error $\mathbf{y} - h_w$ at the output layer is determinable. On the other hand, the error in the hidden layers seems not very well known, since the learning data do not indicate what the value of the hidden nodes should be. In this context, it is possible to backpropagate the error from the output layer to the hidden layers, by a gradient derivation of the global error. Thus, for the output layer, the rule that is updated weights is identical to the equation if below. This is why, there will be several output units, suppose (Err$_i$) will be the i-th element of the error vector y$-h_w$.

Model théoriques de réseau neurone : Is easy to define the error modified by:

$$\Delta_i = Err_i * g'(in_i) \qquad \text{eq.1}$$

To update the connections between input and hidden units, a quantity similar to the error term should be defined for the output nodes [12]. Thus, it is possible to give the following weights:

$$W_{j,i} \leftarrow W_{j,i} + \propto \times E_j \times \Delta_i$$

In addition, it is possible to start the reverse propagated error. The idea is that each hidden node j is responsible for a fraction of the error Δi in each of the input nodes to which it is connected. As a result, the values Δi are divided according to the strength of the link between the hidden node and the output node and are retro propagated to provide the values Δj for the hidden layer in:

$$\Delta_j = g'(in_i) \sum_i W_{j,i}\Delta_i \qquad \text{eq.2}$$

the hidden layer are retained identical to the update rule for the output layer, which allows to write:

$$W_{j,i} \leftarrow W_{j,i} + \propto \times E_j \times \Delta_i ,$$

principles, it is possible to define the quadratic error as follows:

$$E = \frac{1}{2}\sum(y_i - E_i)^2 \qquad \text{eq.3}$$

Where, the sum that is presented for the nodes of the output layer. To obtain the gradient relative to a weight $(w_{j,l})$ it is possible to expand the activation ai since no other term of the summation is affected by $(w_{j,l})$:

$$\frac{\partial E}{\partial w_{j,i}} = (y_i - E_i)\frac{\partial E_i}{\partial w_{j,i}}, \qquad \text{eq.4}$$

$$= -(y_i - E_i)\frac{\partial g(in_i)}{\partial w_{j,i}}$$

$$= -(y_i - E_i)g'(in_i)\frac{\partial in}{\partial w_{j,i}},$$

$$= -(y_i - E_i)g'(in_i)\frac{\partial g}{\partial w_{j,i}}(\sum_j w_{j,i}E_j),$$

$$= -(y_i - E_i)g'(in_i)E_j,$$

$$= -E_j\Delta_i.$$

Δ_i being defined as before. To get the gradient relative to the weights $w_{k,j}$ connecting the input layer to the hidden layer, keep all summation on i because each output value ai can be affected by the changes in $w_{k,j}$. Activations must also be developedE_j. We will show the calculation in detail because it is interesting to see how the gradient performs the backscatter in the network. Because, it has been proposed the error gradient retro propagation learning algorithm which is based on the error gradient calculation, in order to minimize the error at the network output, as follows:

$\frac{\partial E}{\partial w_{k,j}} = -\sum_i (y_i - E_i)\frac{\partial E_i}{\partial w_{k,j}} = -\sum_i (y_i - E_i)\frac{\partial g(in_i)}{\partial w_{k,j}}$, eq.5

$= -\sum_i (y_i - E_i)g'(in_i)\frac{\partial in_i}{\partial w_{k,j}} = -\sum_i \Delta_i \frac{\partial}{\partial w_{k,j}}(\sum_j w_{j,i}E_j)$,

$= -\sum_i \Delta_i w_{j,i}\frac{\partial E_j}{\partial w_{k,j}} = -\sum_i \Delta_i w_{j,i}\frac{\partial g(in_j)}{\partial w_{k,j}}$,

$= -\sum_i \Delta_i w_{j,i}\, g'(in_j)\frac{\partial in_j}{\partial w_{k,j}}$,

$= -\sum_i \Delta_i w_{j,i}\, g'(in_j)\frac{\partial}{\partial w_{k,j}}(\sum_k w_{k,i}E_k)$,

$= -\sum_i \Delta_i w_{j,i}\, g'(in_j)E_k = -E_k\Delta_j$.

Where, Δj is defined as before. Next, it is important to define the accuracy of the R.N.A model which is evaluated using the following formulae (Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE).

$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - \hat{x}_i|$ eq.6

$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2}$ eq.7

$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{x_i - \hat{x}_i}{x_i}\right| * 100\%$ eq.8

Where x_i is the measured value and $\hat{x}_i$ is the predicted value.

To conclude this part of modelling, they are developed through the learning algorithm and their corresponding mathematical models. Thus, in the objective which has been defined, this modelling aims at the proposal of a prediction method capable of taking into account the uncertainty in the parameters which affect the prediction. Once defined, this model can be applied at a site in the north of the Mauritanian coast (Ballawack).

Sigmoid mathematical function : The activation function converts the net value into the neuron input. In this case, the standard sigmoid function which is a logarithmic sigmoid function is proposed. Indeed, it is a non-linear activation differentiable in accordance with S(x) of R to (0,1) defined by Logistic sigmoid: and it is proposed as follows in equation:

$f(x) = \frac{1}{1 + e^{-x}}$ eq.9

In addition, it is possible to develop a network of artificial neurons capable of predicting or modelling an energy system through network data (inputs). These data are system inputs and outputs to be predicted to help the network learn the system. Thus, the learning of the network will go through the following 5 steps:

- Weighting of inputs by parameters called weight(W),
- Summation of weighted inputs,
- Calculation of neuron response by activation function,
- Calculation of the error between the theoretical output and that calculated by the R.N.A,
- Weight modification to minimize error by a specific mathematical algorithm called learning algorithm.

In conclusion of this part, it is possible to say that this activation function that was chosen represents the transfer function that will connect the weighted summation to the output signal. Indeed, the proposed mathematical model illustrates one of the most commonly used activation functions. It is known as sigmoide function.

Theoretical Wind Profile Model: The meteorological records show that the relative growth of the wind speed with altitude relative to the ground in a homogeneous site, varies from one point to another is given by the law of vertical extrapolation:

$\frac{v_1}{v_2} = \left[\frac{h_2}{h_1}\right]^n$ eq.10

v2 = Wind speed at h2 height, v1 = Known wind speed at h1 height,

h2 = Height at which the wind speed is to be estimated, h1 = Reference height, n = value that depends on the roughness existing on the location. n is also, a coefficient varying from 0.10 to 0.4. The lowest values correspond to the windy areas and the highest to the low intensity areas.

Approche algorithme : Le modèle va être réalisé à travers Toolboxnntwork de Matlab. Sans oublier de noter que la génération du modèle *Simulink* sera effectuée à l'aide de la fonction *ntstool*.
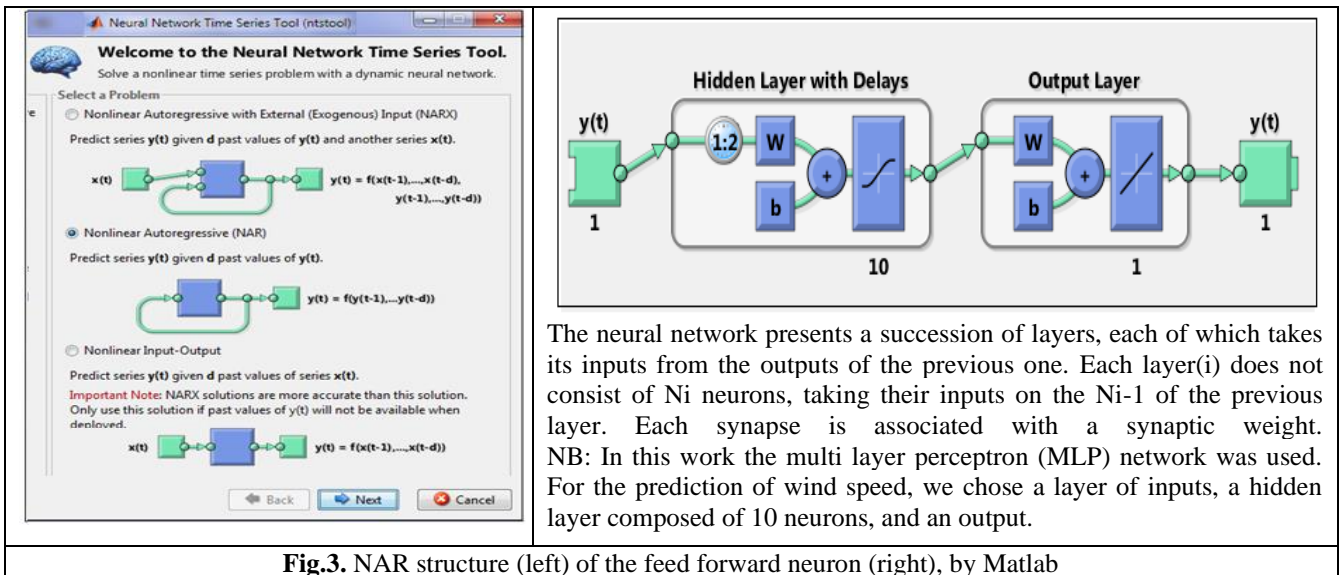
The neural network presents a succession of layers, each of which takes its inputs from the outputs of the previous one. Each layer(i) does not consist of Ni neurons, taking their inputs on the Ni-1 of the previous layer. Each synapse is associated with a synaptic weight. NB: In this work the multi layer perceptron (MLP) network was used. For the prediction of wind speed, we chose a layer of inputs, a hidden layer composed of 10 neurons, and an output.

Fig.3. NAR structure (left) of the feed forward neuron (right), by Matlab

For the structure of the neural network feed-forwad of Figure.3 (right) is presented a succession of layers (an input layer, a hidden layer composed of 10 neurons, and an output) to realize the prediction of the wind speed. Thus, for prediction through Matlab, the Simulink model is proposed in Figure 4.
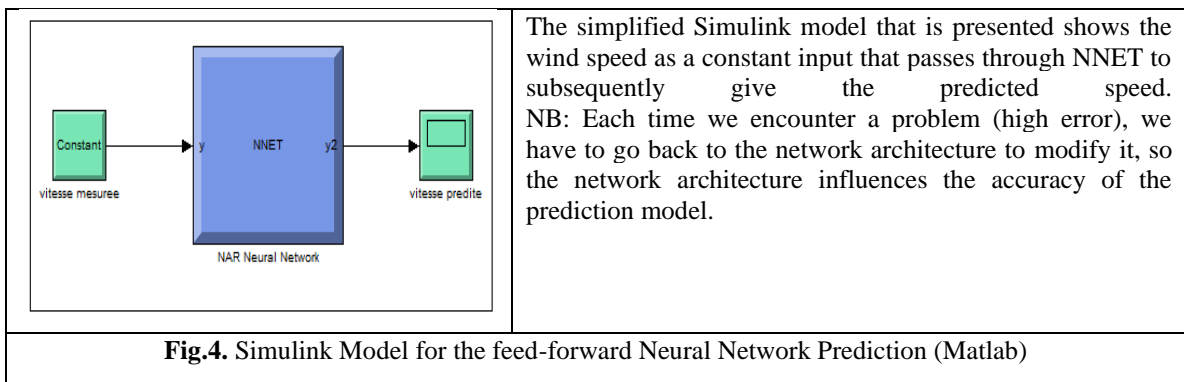

The simplified Simulink model that is presented shows the wind speed as a constant input that passes through NNET to subsequently give the predicted speed. NB: Each time we encounter a problem (high error), we have to go back to the network architecture to modify it, so the network architecture influences the accuracy of the prediction model.

Fig.4. Simulink Model for the feed-forward Neural Network Prediction (Matlab)

System design by data preparation: For system design, it is available:

- An efficient data collection system that will provide quality data. This device is carried out through wind speed measurement stations that have been installed on site by the LRAER (Laboratory for Applied Research). So, there is a database of wind on the North Shore that was effective between [03/07/2015, 03/05/2016], An efficient presentation of the data and compatible with subsequent processing steps,

- An efficient presentation of the data and compatible with subsequent processing steps.

It should be noted in addition, that it is proposed an approach based on the computer tool to model and simulate the neural network through Matlab. Thus, Matlab is used to carry out programming in the modeling and simulation of neural networks. Through:

- A training of the model,

- The realization of a program for the neural network feed- forwad.

Model drive by Matlab: During the drive process the weights are adjusted in order to build for each measured input into the network a predicted output. A number of 14 training algorithms were invested for the development of the MLP network, Levenberg - Marquardt back propagation (trainlm) training algorithm and Hyperbolic Tangent Sigmoid (logsig) transfer function (see figure 5) were used. In this sense, it is important to note the realization of a Simulink prediction model (see figure.5).
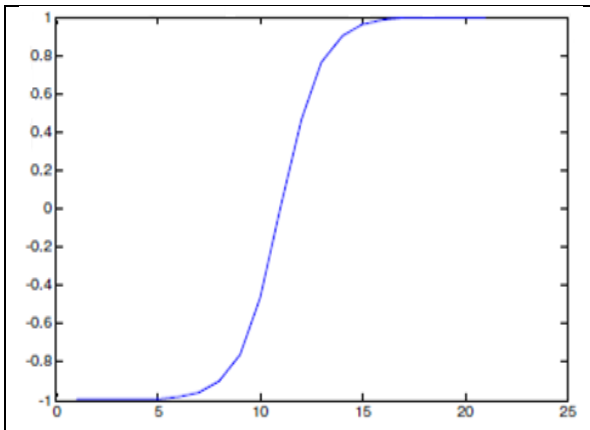
Retrieval Number: 100.1/ijitee.K78530991120
DOI: 10.35940/ijitee.K7853.1091220
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

372

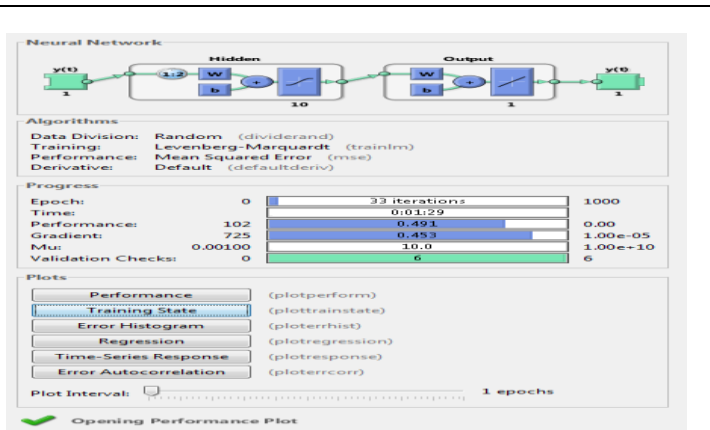| **Fig.5. logsig Transfer function (Matlab)** | **Fig.6. Result under MLP network Training Window (Matlab)** |

Network prediction network program feed-forwad (Matlab):

```
% Solve an Autoregression Time-Series Problem with a NAR
Neural Network
% Script generated by NTSTOOL
% Created Thu Dec 01 18:47:44 PKT 2018
%This script assumes this variable is defined:
% inputsvitesse - feedback time series.
%target Seriestonndata (inputsvitesse,false,false)
% Create a Nonlinear Autoregressive Network
feedbackDelays = 1:2;
hiddenLayerSize = 10;
net= narnet(feedbackDelays,hiddenLayerSize);
% Choose Feedback Pre/Post-Processing Functions
% Settings for feedback input are automatically applied to
feedback output
% For a list of all processing functions type: help nnprocess
net.inputs{1}.processFcns                              =
{'removeconstantrows','mapminmax'};
% Prepare the Data for Training and Simulation
% The function PREPARETS prepares timeseries data for a
particular network,
% shifting time by the minimum amount to fill input states and
layer states.
% Using PREPARETS allows you to keep your original time
series data unchanged, while
% easily customizing it for networks with differing numbers of
delays, with
% open loop or closed loop feedback modes.
[inputs,inputStates,layerStates,targets]                    =
preparets(net,{},{},targetSeries);
% Setup Division of Data for Training, Validation, Testing
% For a list of all data division functions type: help nndivide
net.divideFcn = 'dividerand';  % Divide data randomly
net.divideMode = 'time';  % Divide up every value
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
% Choose a Training Function
% For a list of all training functions type: help nntrain
net.trainFcn = 'trainlm';  %Levenberg-Marquardt

% Choose a Performance Function
% For a list of all performance functions type: help
nnperformance
net.performFcn = 'mse'; % Mean squared error
perform(net,trainTargets,outputs)
valPerformance = perform(net,valTargets,outputs)
testPerformance =
```
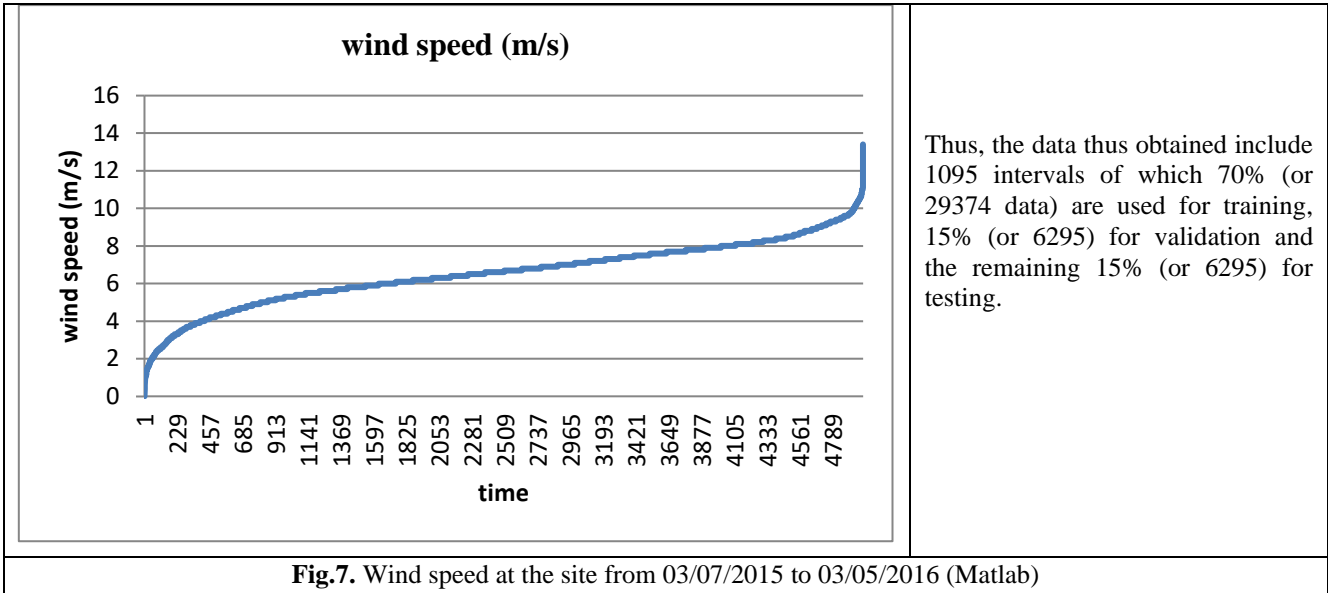
```
% Choose Plot Functions
% For a list of all plot functions type: help nnplotnet.plotFcns=
{'plotperform','plottrainstate','plotresponse', ...
'ploterrcorr', 'plotinerrcorr'};
%          Train          the          Network[net,tr]          =
train(net,inputs,targets,inputStates,layerStates);
% Test the Network
outputs = net(inputs,inputStates,layerStates);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs)
% Recalculate Training, Validation and Test Performance
trainTargets = gmultiply(targets,tr.trainMask);
valTargets = gmultiply(targets,tr.valMask);
testTargets = gmultiply(targets,tr.testMask);
trainPerformance =
perform(net,testTargets,outputs)
% View the Network
view(net)
% Plots
% Uncomment these lines to enable various plots.
%figure, plotperform(tr)
%figure, plottrainstate(tr)
%figure, plotresponse(targets,outputs)
%figure, ploterrcorr(errors)
%figure, plotinerrcorr(inputs,errors)
% Closed Loop Network
% Use this network to do multi-step prediction.
% The function CLOSELOOP replaces the feedback input with a
direct
% connection from the outout layer.
netc = closeloop(net);
[xc,xic,aic,tc] = preparets(netc,{},{},targetSeries);
yc = netc(xc,xic,aic);
perfc = perform(net,tc,yc)
% Early Prediction Network
% For some applications it helps to get the prediction a timestep
early.
% The original network returns predicted y(t+1) at the same time
it is given y(t+1).
% For some applications such as decision making, it would help
to have predicted
% y(t+1) once y(t) is available, but before the actual y(t+1)
occurs.
% The network can be made to return its output a timestep early
by removing one delay
% so that its minimal tap delay is now 0 instead of 1.  The new
network returns the
% same outputs as the original network, but outputs are shifted
left one timestep.
nets = removedelay(net);
[xs,xis,ais,ts] = preparets(nets,{},{},targetSeries);
ys = nets(xs,xis,ais);
closedLoopPerformance = perform(net,tc,yc)
```
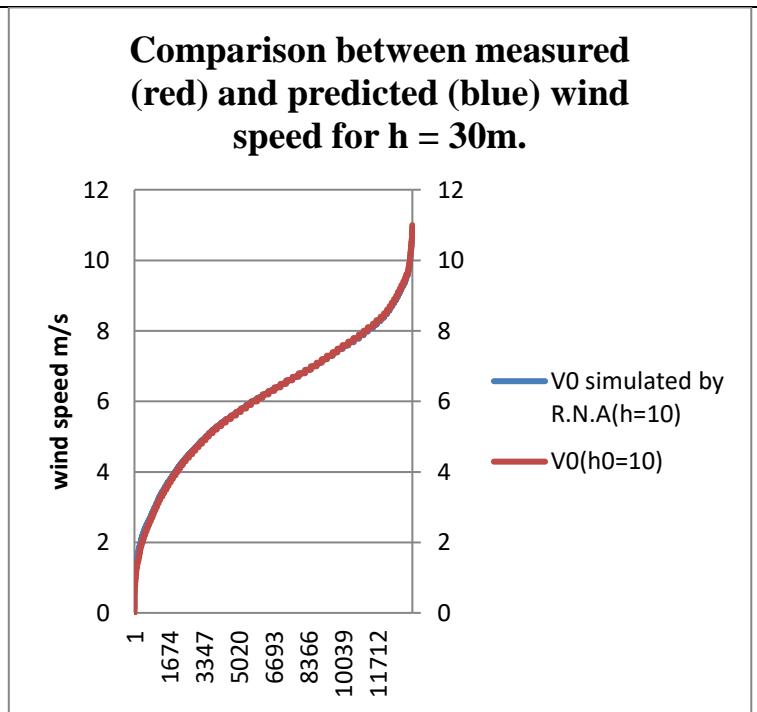
Retrieval Number: 100.1/ijitee.K78530991120
DOI: 10.35940/ijitee.K7853.1091220
Journal Website: www.ijitee.org

373

Published By:
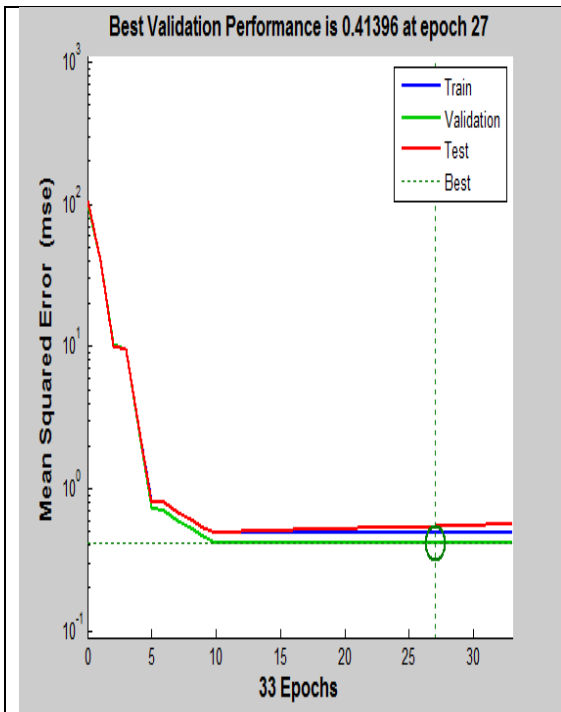*Blue Eyes Intelligence Engineering*
& Sciences Publication

III.   RESULTS AND DISCUSSION

Performance evaluation: Figure.7 shows the wind speed at the site measured at the measuring mast in m/s. It shows the behaviour of the hourly wind speed. In addition, from these raw wind speed data, it is possible to obtain a monthly wind per speed data interval with the min-max and mean approaches.

Thus, the data thus obtained include 1095 intervals of which 70% (or 29374 data) are used for training, 15% (or 6295) for validation and the remaining 15% (or 6295) for testing.

Fig.7. Wind speed at the site from 03/07/2015 to 03/05/2016 (Matlab)

Wind Speed Prediction Performance: Thus, Figure.8 illustrates the performance of the prediction model and shows that the prediction model has achieved the learning of the energy system, as it made it possible to generalise the data not encountered of the system with small simulation errors (see the part (Zoom in figure.9 a gives differences between the different traits (blue accentuated, light green, red and dashed) close to zero.

Fig.8. Neural network training performance for wind speed prediction for h = 10m

Fig.9. Comparison between measured (red) and predicted (blue) wind speed for h = 10m.

Based on the results of this simulation model, it can be seen that this model has achieved good accuracy in the estimated wind speed values. Figure.9, on the other hand, shows the comparison between predicted and measured values. The two lines (accentuated blue and red) are almost confused.

Prediction of wind speed for different heights: In this case, it is an analysis through simulations as a function of h (when roughness is constant) or as a function of roughness (when h is constant). Thus, Figure.10 above represents the different wind speed simulation curves in the roughness 0.12m Ballawack site at heights of 10, 30, 50, 60 and 70 m. This figure shows that the wind speed increases with the attitude in this area which is characterized by a relatively low roughness (Atlantic coast). Which is consistent with the physical understanding of the phenomenon, the higher the wind turbine rises, the more it will encounter higher wind speeds.
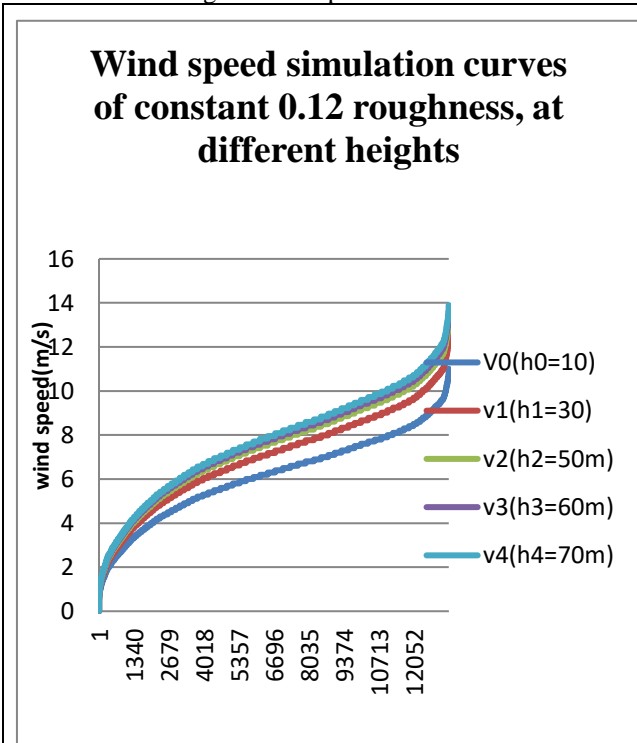

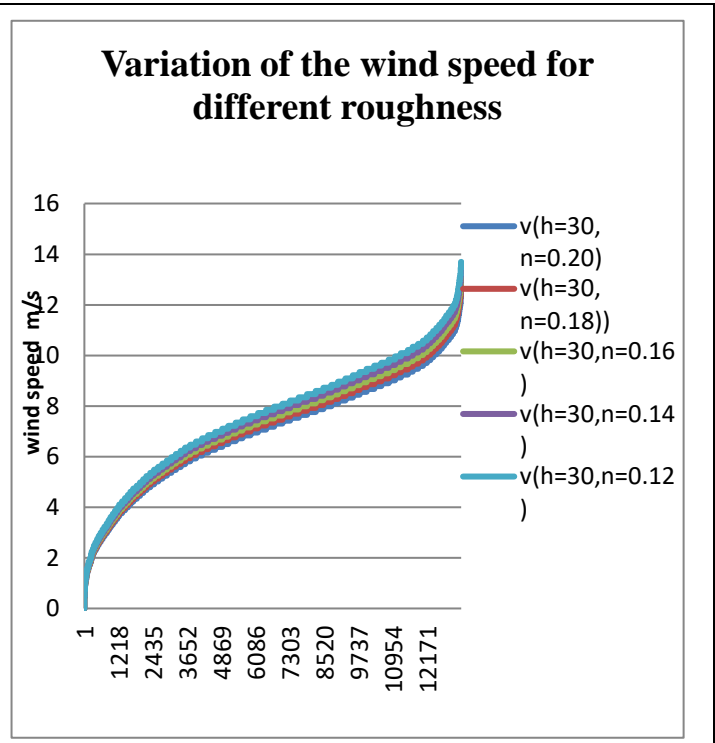| Fig.10. Wind speed simulation curves of constant 0.12 roughness, at different heights	Fig.11. Variation of the wind speed for different roughness (Matlab) and a height h = 30m.

Figure.11, on the other hand, shows the change in wind speed at the Ballawakh site for different roughnesses (0.12, 0.14, 0.16, 0.18 and 0.20). Indeed, the figure shows that the higher the roughness at the site, the lower the wind speed. Prediction of the power of production: In this case, it is an analysis of the electric production which carried out from the simulations according to different heights, when the roughness is constant or following several roughnesses when the height is constant.

Power at different heights:

Predicted power according to different heights
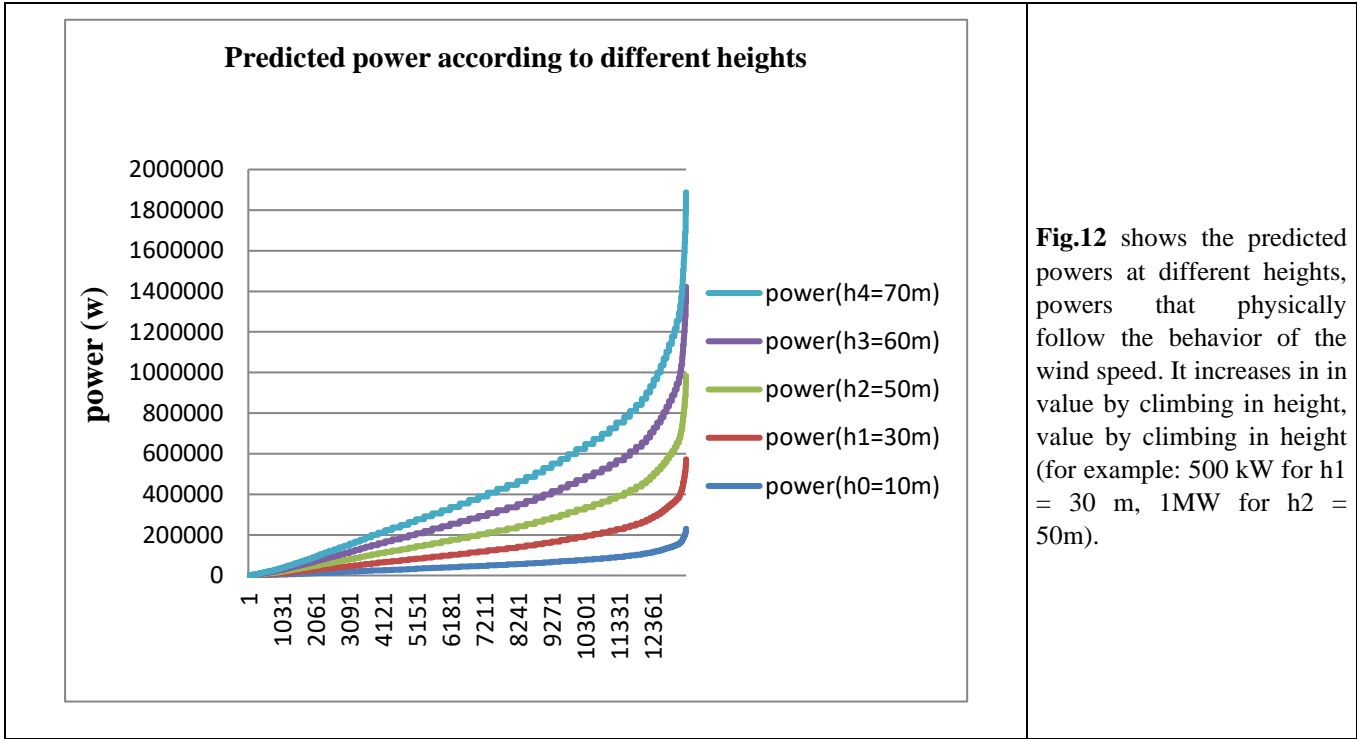
Fig.12 shows the predicted powers at different heights, powers that physically follow the behavior of the wind speed. It increases in in value by climbing in height, value by climbing in height (for example: 500 kW for h1 = 30 m, 1MW for h2 = 50m).

Fig.12. présentation des courbes de puissances en fonction des différentes hauteurs

Power with different roughnesses:

Predicted power according to different roughnesses
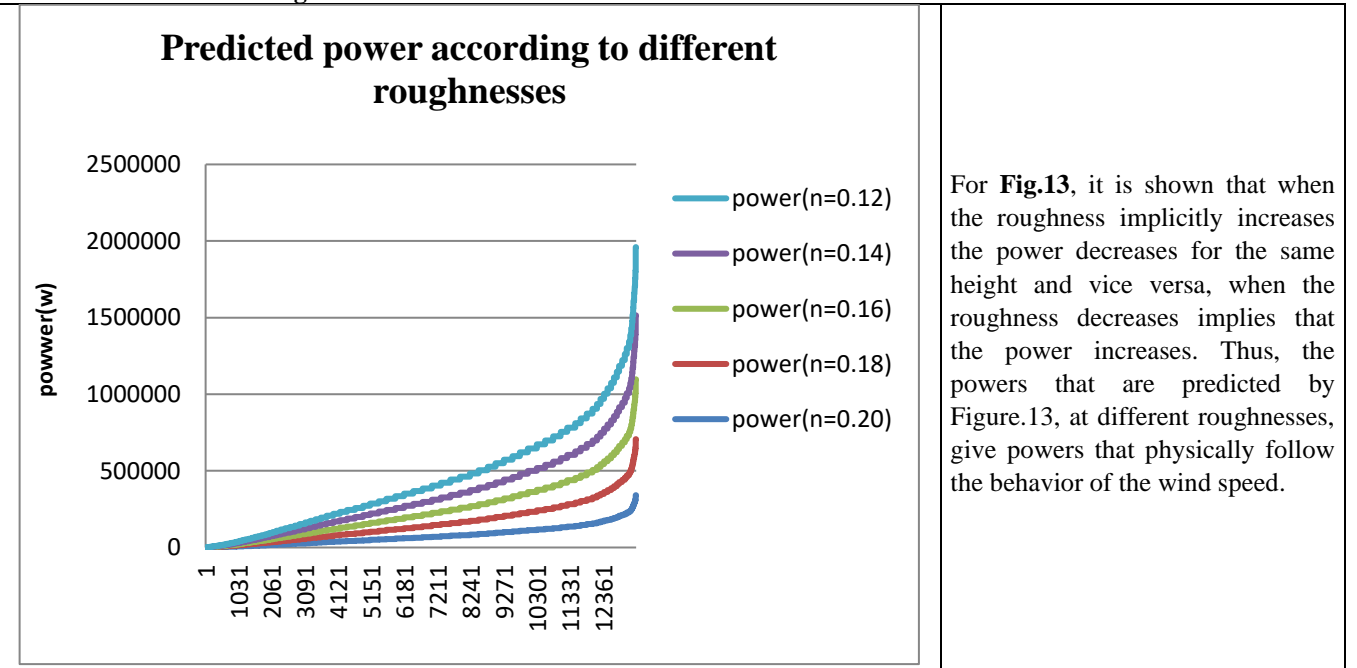
For **Fig.13**, it is shown that when the roughness implicitly increases the power decreases for the same height and vice versa, when the roughness decreases implies that the power increases. Thus, the powers that are predicted by Figure.13, at different roughnesses, give powers that physically follow the behavior of the wind speed.

Fig.13. Predicted power to different roughness and for a height (h = 30 m constant)

Improved performance by neural network feed- forwad: It has been tried to demonstrate that the neural network feed-forwad can cope with an over-adjustment (over-training) under certain conditions for the data set. Indeed, an over-adjustment occurs when the network memorizes drive patterns. In this case, even if the prediction error on the drive stand is small, it may be important on the test set. It should also be said that the number of input characteristics (i.e., input neurons), the number of hidden neurons and the number of training samples are all important factors that can play a role in adjustment [10].

To avoid over-adjustment, a validation set (a fixed set of samples not included in the learning set) can be used to make an "early stop". To do this, the basic technique of early arrest proceeds as in [11 and 12]. It is then possible to say in accordance with Figure 9 that the wind speed prediction model that was determined is a model that has extinguished the best performance in 27th iteration.

To better illustrate the error, it is proposed as part of this discussion, the table which contains values predicted by the network model feed-forwad. It was then compared with the actual measured values.

Nevertheless, it is important to note in this part of the presentation that the prediction model has advantages and disadvantages:

- In terms of benefits: the system is robust compared to noise data. It allows a very high tolerance to uncertainty. It allows to model large varieties of behaviors. It has a choice of types, architecture and activation function of various networks. It is easy to operate because it does not require a thorough understanding and in the end it has a capacity for generalization,
- At the level of disadvantages: There is a level of difficulty in explaining the results related to the opacity of neural networks which prevents a relevant analysis of the solutions obtained,
- It should also be noted that despite a solid theoretical basis, the choice of the network often belongs to the user because there is no appropriate and recognised guide. In the end, another drawback arises in terms of the not optimal architecture. Because, there are still no means to define the optimal architecture of the neural network.

IV. CONCLUSION

In this work it has been shown that networks of artificial neurons feed-forwad have achieved a good prediction accuracy of the meteorological variable (wind speed and wind speed depending on the operating heights of wind systems). This makes it possible to respond in the short term and in the long term in order to find a model capable of solving the problem linked to the variability of wind production which is the main problem that hinders the evolution of wind systems. Within the framework of this conclusion, two recommendations can be drawn, following the non-linear nature which can trap in a local minimum where the performances of the networks are clearly under optimal. To avoid this trap in a local minimum, it is proposed:

- Modifying the network learning step to push the network out of the local minima and at the same time, we manage to adjust the size of the error area,
- The training of the same network from several initial choices of weight, to then keep only the best of them.

Finally, it should be noted that the main difficulties encountered in the problem of predicting wind speed and the influence of parameters such as the profile and roughness encountered in the course of this work are during the implementation of the modelling or prediction.

ACKNOWLEDGEMENTS

REFERENCES

1. Murwantara, Pujianto Yugopuspito, Rickhen Hermawan, Comparison of machine learning performance for earthquake prediction in Indonesia using 30 years historical data, Vol 18, No 3,June 2020, TELKOMNIKA (Telecommunication, Computing, Electronics and Control) ISSN: 1693-6930, e-ISSN: 2302-9293;
2. Sana K. Abdulhassan, Optimal Location of Unified Power Flow Controller (UPFC) Genetic Algorithm Based, Vol 11, No 2, International Journal -IJPEDS.
3. *Nur Haizum Abd Rahman, Muhammad Hisyam Lee, Artificial neural network forecasting performance with missing value imputations, IAES International Journal of Artificial Intelligence (IJ-AI), Vol 9, No 1, IAES International -IJ-AI, ISSN/e-ISSN 2089-4872/2252-8938.*
4. *Pratheek I, Joy Paulose, Prediction of Answer Keywords using Char-RNN, International –IJECE,Vol. 9, No. 3, June 2019, pp. 2164-2176, ISSN: 2088-8708, DOI: 10.11591/pp2164-2176.*
5. Bunn, D.. 2000. « Forecasting loads and prices in competitive power markets ». *In Proceedings of the IEEE,* vol 88, no2, pp.163-169
6. Hahn, H., Meyer-Nieberg, S. and Pickl, S. (2009) Electric Load Forecasting Methods: Tools for Decision Making. European Journal of Operational Research, 199, 902-907.
7. P Pinson, *Very short-term* probabilistic *forecasts*, which are essential for an optimal manage input to the (*offshore*) wind farm controllers themselves with typical time steps between 5, The methodology is *applied* in Section 5 ... Energ., *88, 4087–4096.*
8. Kusiak, A, Zheng, H and Z Song (2009). Short-term prediction of wind farm power: A data mining approach. IEEE Transactions on Energy Conversion, 24(1):125-136.
9. Kusiak, A, Zheng, H and Z Song (2009). Wind farm power prediction: a datamining approach. Wind Energy, 12(3):275-293.
10. EPFL, November 2004, W. Gerstner P. Comon Thomson-Sintra, Classification supervisée par réseaux multicouches, Parc de Sophia Antipolis,BP 138, F-06561
11. Landajo, M., de Andrés, J., 2007. Robust neural modeling for the cross-sectionalanalysis of accounting information. European Journal of Operational Research177 (2), 1232–1252
12. Ak, R, Li, Y, Vitelli, V and E Zio (2013). A genetic algorithm and neural network technique for predicting wind power under uncertainty. In Prognostics and System Health Management Conference PHM-2013
13. Ahmed A, Khalid M. An intelligent framework for short-term multi-step wind speed forecasting based on Functional Networks. Appl Energy 2018;225(April):902–11
14. Potter, C W, Negnevitsky, W (2006). Very Short-Term Wind Forecasting for Tasmanian Power Generation.*IEEE Trans. Power Syst.*, 21(2), 965–972.
15. Abreu, L.. 2008. « Wind power generation and dispatch in competitive power markets ». PhD. dissertation, Chicago, IL., Illinois Institue of Technology
16. «Clean energy progress report ». Abu Dhabi: International Energy Agency (IEA) - 2016
17. CanWEA. 2010. « 2025 - La force du vent: Une stratégie pour le Québec ». In *Canadian Wind Energy Association (CanWEA) 26e annual congress*. Montréal
18. Barbounis, T., Theocharis, J., 2007.A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation. Neurocomputing 70, 1525–1452
19. Abreu, L.. 2008. « Wind power generation and dispatch in competitive power markets ». PhD. dissertation, Chicago, IL., Illinois Institue of Technology
20. «Clean energy progress report ». Abu Dhabi: International Energy Agency (IEA) - 2016
21. CanWEA. 2010. « 2025 - La force du vent : Une stratégie pour le Québec ». In *Canadian Wind Energy Association (CanWEA) 26eannualcongress*. Montréal
22. Barbounis, T., Theocharis, J., 2007.A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation. Neurocomputing 70, 1525–1452
23. Ata, R (2015). Artificial neural networks applications in wind energy systems: a review, Renewable and Sustainable Energy Reviews, 49, 534-562, doi10.1016/j.rser.2015.04.166
24. Cadenas E, Rivera W. Wind speed forecasting in three different regions of Mexico, using a hybrid ARIMA-ANN model. Renew Energy 2010;35(12):2732–8

Retrieval Number: 100.1/ijitee.K78530991120
DOI: 10.35940/ijitee.K7853.1091220
Journal Website: www.ijitee.org

377

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication