Horizontal Extrapolation of Wind Speed Distribution Using Neural Network for Wind Resource Assessment

Nihad AGHBALOU[1,2], Abderafi CHARKI[2], Saida RAHALI ELAZZOUZI[1], Kamal REKLAOUI[1]

[1] University Abdelmalek Essaâdi-Tangier, Morocco

[2] University of Angers-Angers, France

Abstract: ***To evaluate the wind potential on a site for future wind energy project, an accurate representation of the wind speed distribution is required. However, due to the lack of observations, wind engineers are conducted to use some statistical tools to estimate the characteristics of wind by the measurements from a nearby reference or data obtained from a short period. In this work, we aim at applying an information processing paradigm that is inspired by biological neurons, formal neurons, for the assessment of wind speed distribution. Two different learning algorithms are used so as to generate Artificial Neural Network with one hidden layer. Results prove that learning by means of Bayesian regularization, in comparison with Levenberg–Marquardt learning algorithm, gives the best performance. In addition, the proposed network allows significant results in horizontal wind extrapolation.***

Keywords: Wind Speed Distribution, Weibull distribution, Energy Performance, Neural Network, Bayesian Regularization, Levenberg–Marquardt

1. Introduction

Wind energy field has known a remarkable deployment in both developed and developing countries. It represents a real alternative among other energy resources (solar, biomass, etc.). Wind climate determination and assessment of potential producible of future wind farms constitute a joint primary steps in the techno-economic study of all wind projects. Then, the choice of an appropriate turbine is required to ensure maximum reliability in terms of electricity. Meteorological parameters (wind, temperature, pressure, etc.) are difficult to be forecasted and so uncertain in time and space. The need to develop robust and reliable tools to evaluate a site with good accuracy facilitates the task and supports operators of electric grids to integrate more wind energy into their energetic portfolios. With the advent of the paradigms of artificial intelligence and machine learning, modeling the behavior of stochastic and complex processes, such as wind and its energy conversion, became feasible with reasonable accuracy.

In literature, we find that artificial intelligence methods were widely used for forecasting and predicting wind energy and wind speed. They include artificial Neural Networks[1], the Adaptive Neuro-Fuzzy Inference Systems[2], Separators Wide Margin[2], Genetic Algorithm[2], Neuro-Fuzzy Networks[2],and Neural Network with RBF (Radial Basis Function) [3], other methods are given in references [4], [5]and [6]. The Neural Networks are assemblies of computing units, strongly connected forming computational models able to solve a very complex problem: classification; pattern recognition; time series prediction; etc. It is frequently an ideal solution for “black box” modeling, when we have data and we don’t know the rules that govern the phenomenon that we seek to model. However, Artificial Neural Networks (ANNs) have the disadvantage of overfitting and generalization.

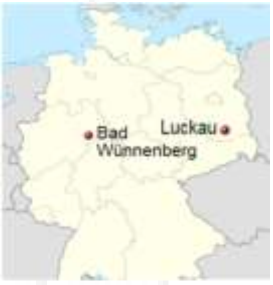

Figure 1: Location of sites under study

In this work we propose to apply ANNs using two different algorithms in the learning phase in order to estimate the distribution of wind speed in two sites: Wuennenberg, situated in the West of Germany (Longitude: 8.7E Latitude: 51.517N Altitude: 281m) and Luckau, situated in East of Germany (Longitude: 13.667E Latitude: 51.833N Altitude: 75m). Data, available on line [http://renknownet2.iwes.fraunhofer.de/pages/wind_energy_0150.htm], are instantaneously recorded values every 5 minutes at 10 m above ground level and cover the period starting from 1/4/2003 to 31/3/2004.

2. Methods and Materials

2.1. Artificial Neural Network

The ANNs are classified as an artificial intelligence tool. As the biological neurons, the ANNs are dynamic systems that learn by examples. They are formed by formal neurons that interact. These interactions are defined by the connection weights, which in general change intensities during the learning phase. This learning consists in modifying the connection weights until stabilization of the network. That means, until the weight does not change more than a tiny way. ANNs have a problem of setting the number of the neurons in the hidden layers, and then overfitting and generalization after the learning process[7], which implicates the need to combine them with other techniques for improvement.

- **Levenberg-Marquardt optimization based Back-propagation (LM-BP)**

Backpropagation algorithm is basically the most usedalgorithm for training multilayer networks in different modeling issues. The backpropagation learning law is an approximate steepest descent algorithm[8], in which the aim is to minimize the performance index, mean square error. The two main drawbacks of this algorithm are the slow convergence and the sensitivity to get stuck in local minima[9]. The Levenberg-Marquardt algorithm is a variation of Newton's method that was designed for minimizing functions that are sums of squares of other nonlinear functions [8]. Hagan et al. [8] are found that the Marquardt modification to the backpropagation is very well suited to neural network training where the performance index is the mean squared error. Hereafter, the iterations of this algorithm are summarized:

a. Present all inputs (1) to the network and compute the corresponding outputs (2) of the layers :

$$a^0 = (x_1, x_2, x_3 \dots) \qquad (1)$$

$$a^{k+1} = f^{k+1}(w^{k+1}a^k + b^{k+1}) \text{ for k=0,1..M-1} \qquad (2)$$

M is the number of the layer in the network, $f^k(\, . \,)$ is the transfer function of the *k* layer, and a^0 is the input vector of the network. w^k and b^k are in succession weight and bias in the layer *k*.

b. Compute the sum of squared errors over all inputs:

$$v(\underline{w}) = \frac{1}{2}\sum_{q=1}^{Q}(t_q - a_q)^2 = \frac{1}{2}\sum_{q=1}^{Q} e_q^2(\underline{w}) \qquad (3)$$

Q is the number of the inputs/targets (t_i/a_i) in the training set. $\underline{w}$is the weights vector.

c. Compute the Jacobian matrix:

$$J(\underline{w}) = \frac{\partial(e_1, \dots, e_Q)}{\partial(w_1, \dots, w_Q)} \qquad (4)$$

d. Compute the update term of the weight as follow :

$$\Delta\underline{w} = \left[J^T(\underline{w})J(\underline{w}) + \mu I\right]^{-1} J^T(\underline{w})e(\underline{w}) \qquad (5)$$

e. Recompute the sum of squared errors using adjusted weights:

$$\underline{w} = \underline{w} - \Delta\underline{w} \qquad (6)$$

f. If this new sum of squares is smaller than that computed in step b, reduce the Levenberg's damping factor *μ*, let $\underline{w}$=$\underline{w}$-Δ$\underline{w}$and go back to step a. If the sum of squares is not reduced, augment *μ* and go back to step d. The algorithm converges if the condition on error or on the number of iterations is achieved.

- **Bayesian regularization based LM-BP**

The regularization technique aims to force the network to converge to a set of weights with smaller values of bias, so that the network response is smoother and less likely to over-adjustment [10]. It is also an important technique that used to improve generalization capacity of neural network without modifying the number of neurons of the network, just by adding a weight decay function (7) to the sum squared error performance (8) of the neural network[10] to force the resulting function to be smooth.

$$E_w = \sum_{i=1}^{n} w_i^2 \qquad (7)$$

$$E_d = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \qquad (8)$$

N: number of learning samples.
n: number of weights w_i in the network
y_i and $\hat{y}_i$ are respectively the target and the estimated values of the network.

The new objective function of network becomes:

$$F = \alpha E_W + \beta E_D \qquad (9)$$

Where *α* and *β* are the decay rates or the regularizing constants. Bearing in mind that the aim of the learning process is to minimize the objective function *F*, by setting the weights at the optimal values to have a good generalization without overfitting, the optimal values of *α* and *β* are also required to be optimal. In fact, if *α*<<*β*, the training algorithm places the most weight and the errors are too small. If α >>β, the magnitude of the weights will be smaller and the network response smoother[10].

According to the framework developed by David MacKay[11], the selection of optimal parameters of this objective function can be considered as an inference Bayesian problem. Maximizing the posterior probability of regularization parameters assuming that the prior density is uniform given the data *D* and the neural network model *M*, is achieved by maximizing the likelihood function *P(D/α,β,M)*.

$$P(\alpha, \beta/D, M) = \frac{P(D/\alpha, \beta, M)P(\alpha, \beta/M)}{P(D/M)} \qquad (10)$$

On the other hand, minimize the objective function *F* is identical to maximize the a posteriori probability density of the weights given the data, the neural network model and regularization parameters.

The optimal values of regularization parameters around the maximum a posteriori estimates of weights w^{MP} are as follows:

$$\alpha^{MP} = \frac{\gamma}{2E_w(w^{MP})} \qquad (11)$$

$$\beta^{MP} = \frac{N - \gamma}{2E_D(w^{MP})} \qquad (12)$$

$$\gamma = n - 2\alpha^{MP} tr(J^T(\underline{w}^{MP})J(\underline{w}^{MP})^{-1}) \qquad (13)$$

*γ*is called the effective number of parameters.

Those results are implemented with Levenberg-Marquardt Back-propagation algorithm by taking (9) objective function, *α* and *β* are initialized randomly and adjusted following (11), (12) and (13) until convergence.

2.2. Methodology proposed

In order to evaluate the capacity of the Neural Network to model the wind speed distribution function, we proceeded to train the network by changing the size of the training data. The learning data consists of geographical characteristics (longitude, latitude and altitude) of the site; standard deviation and mean monthly wind speed at the input of the network. The output of the network consists of the frequencies of the wind speed contained in intervals [I, I+1[for I=0, …, 22. Transfer functions in the hidden layer are sigmoid, and linear in the output layer.

In general, there is not any universal formula that permits setting the number of neurons in hidden layers. The choice is made by test-error method, increasing (constructive method) or decreasing (pruning method) the number of neurons [7]. However, in this study we propose to apply the formula developed by K.G. Sheela and S. N. Deepa in [7]:

$$N_h = (4n^2+3)/(n^2-8) \approx 6.058 \quad (14)$$

n: the number of parameters in network input, equal to 5 in our case of study.

The performances of the configured models are evaluated by calculating root mean square error, as follows:

$$RMSE = \sqrt{\sum_{1}^{n} \frac{(f(v_i) - \dot{f}(v_i))^2}{N_f}} \qquad (15)$$

$f(v_i)$and$\dot{f}(v_i)$are respectively measured and predicted frequencies and N_f=22.

2.3. Comparative methods of wind energy estimation

The wind resource assessment and the producibility determination of future wind farms are paramount steps in any wind project. A small deviation in these assessments may lead to poor estimation of energy efficiency and the investment cost.

To model the distribution of wind speeds at a site over time, several methods are available, gamma law; the lognormal law; etc[12]. The Weibull distribution is one of the classical statistical methods widely used in the evaluation of the wind resource. The Weibull distribution law is defined by:

$$f(v) = \left(\frac{k}{c}\right)\left(\frac{v}{c}\right)^{k-1} \exp\left(-\frac{v}{c}\right)^{k} \qquad (16)$$

v: wind speed (m/s); *c* : scale factor and *k* form factor.Parameters of Weibull distribution can be estimated by several methods, the most frequently employed are : moments methods [13]; least squares method[14]; the estimation of the maximum likelihood [15][16]. The last one will be used in our comparative case study [17].

In the case where is attempted to model the wind speed in a site where there is no enough data, the Measure-Correlate-Predict method (MCP) is often used[18]. The MCP method based on the ratio of the standard deviations, is used to estimate the long-term wind data at the target site from the wind data at the reference site. The relationship between the velocities in the two measurement points is as follows:

$$\hat{V} = aU + b \qquad (17)$$

$$a = \frac{\sigma_{V_s}}{\sigma_{U_s}} \qquad (18)$$

$$b = \overline{U}_S - \overline{V}_S \frac{\sigma_{V_s}}{\sigma_{U_s}} \qquad (19)$$

$\hat{V}$ and *U* are respectively estimated long term wind speeds at the target site and measured wind speeds at the reference site. $\overline{V}_s$and $\bar{U}_s$are the short term mean deviations of wind speeds.σ_{Vs} and$\sigma_{Us}$are measured standard deviations for short term at the target and reference sites, respectively.

In our case of study we consider three months for the short term based on the correlation coefficient between the short term wind speeds in the two sites.

The discrete distribution law of the average power available in the wind, across a unit surface, perpendicular to the wind speed direction can be assessed by equation[19]:

$$P = \frac{1}{2}\rho \sum_{i=1}^{n} v_i^3 f(v_i) dv \qquad (20)$$

n designating the number of measurements, ρ is the air density, v_i is the wind speed (m/s) , $f(v_i)$ is the frequency of v_i.

To summarize our study, we propose the fig.2.

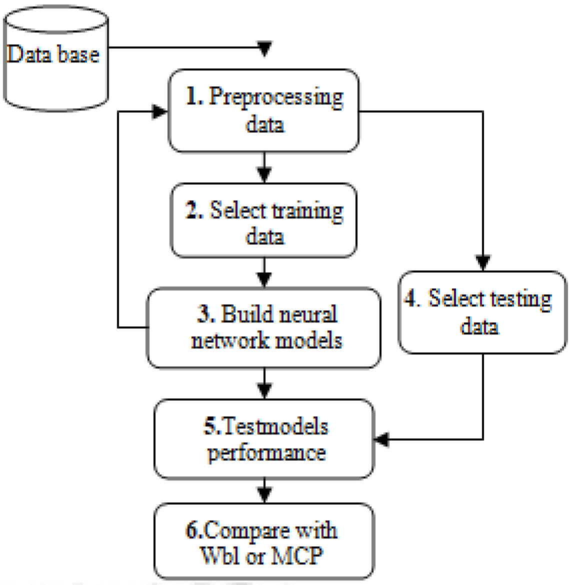

Figure 2: Summary of the study

3. Results and Discussion

The models given in Table.1 are trained and tested using different associated data. At first, the training data set includes total data records between January and September, which makes up 75% of the total data records. In order to get more reliable evaluation and comparison, the models are evaluated by testing data set which was not used during the training process. The testing data set consists of October, November and December records, which is 25% of the total observed data. The Fig.3 illustrates the average training errors made by learning data recorded for nine months in both sites. The figure depicts also the average testing errors for the months which are not used in the training process.

The considered monthly errors are established in Table.2 and Table.3. A comparison with classical model of Weibull highlights that the ANNs models can model the distributionfunction of wind speed given geographical characteristics(longitude, latitude and altitude). Standard deviations and

Table 1: Neural Network's architecture

Model	Training algorithm	Number of hidden nodes
M1	Bayesian regularization	6
M2	Bayesian regularization	7
M3	Levenberg-Marquardt	6
M4	Levenberg-Marquardt	7

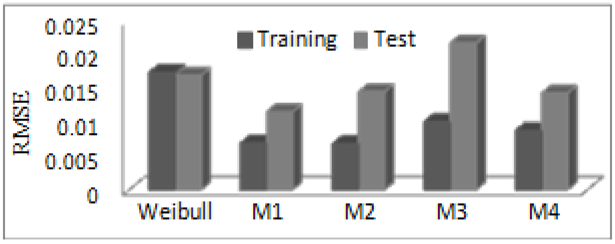

Figure 3: Average training and test errorsconsidering the two sites

meansare estimated monthly for wind speed after training process, more accurately than by conventional Weibull. Indeed, ANN trained with Bayesianregularization M1 and M2 are revealed to be more accurate than ANN trained by means of Levenberg-Marquardt algorithm M3 and M4. Moreover, M1 and M2 are less sensitive to the number of the neurons in the hidden layer.

Table 2: Performance analysis of comparative proposed models at Luckau site

		Weibull	M1	M2	M3	M4
Training	January	0.0172	0.0023	0.0021	0.0043	0.0128
	February	0.0109	0.0024	0.0012	0.0109	0.0045
	March	0.0138	0.012	0.0025	0.019	0.0134
	April	0.0144	0.0043	0.003	0.0123	0.0046
	May	0.0229	0.0123	0.011	0.0094	0.0084
	June	0.0216	0.0049	0.0053	0.006	0.0065
	July	0.018	0.0098	0.0107	0.013	0.0125
	August	0.0177	0.004	0.0045	0.0039	0.0044
	September	0.0178	0.006	0.0043	0.0087	0.0177
Test	October	0.0163	0.0089	0.0099	0.0097	0.0115
	November	0.0139	0.0124	0.0089	0.0193	0.0162
	December	0.0222	0.0114	0.0241	0.0163	0.0117
	Average	0.0175	0.0109	0.0143	0.0151	0.0131

Table 3: Performance analysis of comparative proposed models at Luckau site

		Weibull	M1	M2	M3	M4
Training	January	0.012	0.0127	0.0137	0.013	0.0048
	February	0.0113	0.0022	0.0227	0.0028	0.0057
	March	0.0125	0.0037	0.0016	0.0049	0.0051
	April	0.0172	0.0028	0.002	0.0271	0.0087
	May	0.0271	0.0183	0.0022	0.0094	0.0094
	June	0.0225	0.0026	0.0018	0.0028	0.0138
	July	0.0206	0.0052	0.0162	0.0089	0.0062
	August	0.0179	0.0031	0.002	0.0044	0.0035
	September	0.0193	0.006	0.004	0.0068	0.0064
Test	October	0.0139	0.0141	0.015	0.0221	0.0148
	November	0.0187	0.0088	0.0121	0.0383	0.0182
	December	0.0134	0.0132	0.0136	0.0154	0.0113
	Average	0.0153	0.0120	0.0136	0.0253	0.0148

From the Fig.4, it can be considered that the error values observed by M1 and M2 models are less compared with M3 and M4 models in presenting new data.

Furthermore, those models outperform the conventional Weibull model. Fig.4 and Fig.5 show that the results obtained with the M1 for the typical months are nearer the measured values than those obtained with the Weibull law.

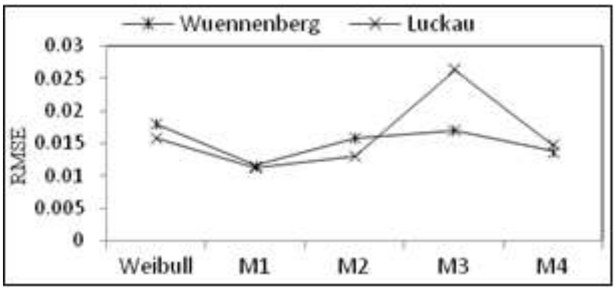

Figure 4: Average errors in testing the comparative modelsconsidering the two sites data for learning

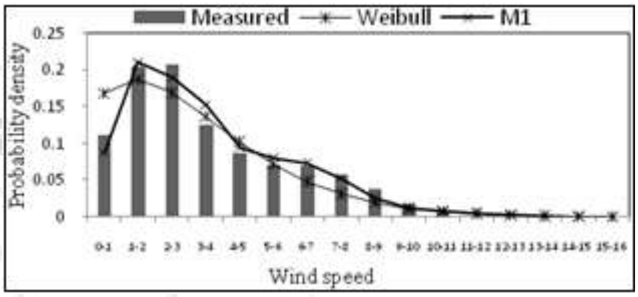

Figure 5: Wind speed distribution laws obtained by M1 andWeibull methods on the site of Luckau on October 2006

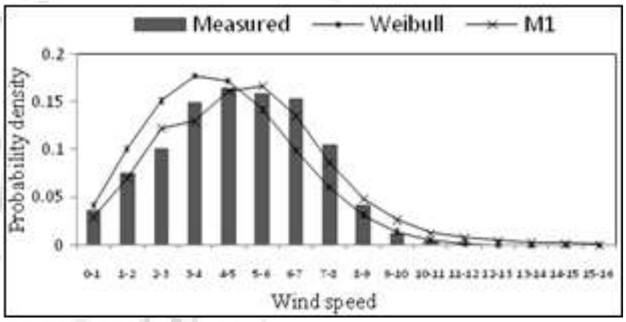

Figure 6: Wind speed distribution laws obtained by M1 and Weibull methods on the site of Wuennenberg on November 2006

The second case of study concerns the extrapolation issue. The collected data is divided into training and testing range. To achieve this, the training data recorded from Wuennenberg site, was used to train the proposed models and predict the long-term wind speedinLuckau site. To predict the long-term wind speed in Luckau, the data used in the input of the networks are monthly means and standard deviations of wind speed at the reference site Wuennenberg and geographical characteristics of the target site Luckau. Table.4 contains information about the different performances of the studied models against the MCP performances. The extrapolated frequencies at Wuennenberg site based on Luckau data are also summarized in Table.5.
As illustrated in Fig. 7, it can be seen that the RMSE value of M1 model is slightly lower than those of M2, M3, M4 and MCP models.

Table 4: Performance analysis at Luckau by extrapolation comparative models from Wuennenberg data

	MCP	M1	M2	M3	M4
January	-	0.0145	0.0197	0.0449	0.0331
February	-	0.0092	0.0032	0.0246	0.0108
March	-	0.0217	0.0175	0.0284	0.0249
April	0.0160	0.0107	0.0108	0.0228	0.0275
May	0.0073	0.0135	0.0162	0.0159	0.0098
June	0.0147	0.0141	0.0403	0.0102	0.0055
July	0.0091	0.0132	0.0068	0.0115	0.0129
August	0.0149	0.0083	0.0495	0.0164	0.0087
September	0.0113	0.0037	0.0070	0.0076	0.0113
October	0.0307	0.0182	0.0250	0.0262	0.0405
November	0.0257	0.0108	0.0134	0.0301	0.0211
December	0.0135	0.0161	0.0093	0.0262	0.0087
Average	0.0159	0.0121	0.0198	0.0185	0.0162

Table 4: Performance analysis at Wuennenberg by extrapolation comparative models from Luckau data

	MCP	M1	M2	M3	M4
January	-	0.0036	0.0050	0.0168	0.0040
February	-	0.0076	0.0211	0.0304	0.0121
March	-	0.0066	0.0164	0.0351	0.0170
April	0.0257	0.0167	0.0231	0.0200	0.0235
May	0.0436	0.0105	0.0172	0.0147	0.0137
June	0.0434	0.0133	0.0084	0.0070	0.0121
July	0.0285	0.0125	0.0087	0.0113	0.0159
August	0.0184	0.0119	0.0154	0.0084	0.0101
September	0.0093	0.0070	0.0157	0.0072	0.0053
October	0.0270	0.0071	0.0082	0.0202	0.0116
November	0.0474	0.0251	0.0349	0.0270	0.0309
December	0.0352	0.0103	0.0178	0.0228	0.0118
Average	0.0309	0.0127	0.0166	0.0154	0.0150

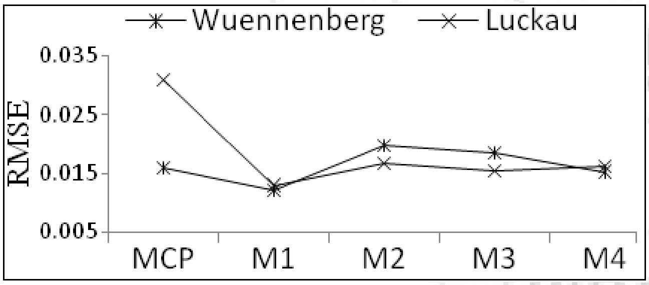

Figure 7: Average errors evaluated by applying extrapolation comparative models

The discrete distribution law of the average power available in the wind, across a unit surface perpendicular to the wind speed direction (equation 20) is applied to determine the monthly available power in the extrapolated wind speeds.Fig.7. and Fig.8.show the contribution of each candidate model.

Obviously, it is revealed that the values achieved with the help of the proposed Neural Network model M1 are closer to the results achieved with the measured values.

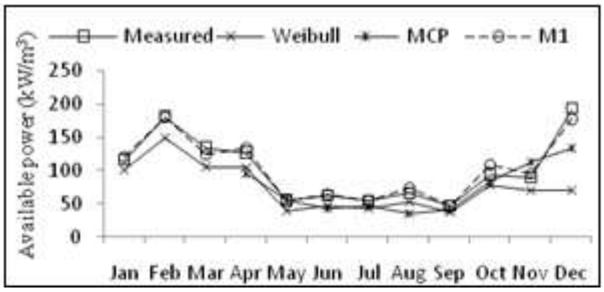

Figure 8: Wind power by swept area at Luckau by extrapolation methods from Wuennenberg data

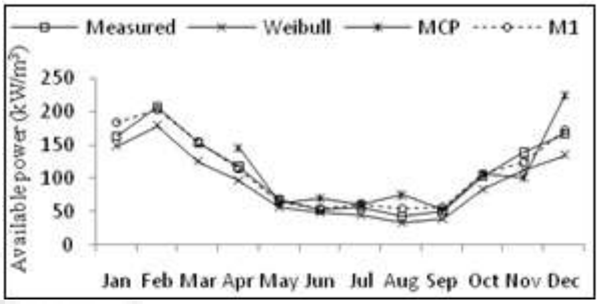

Figure 9: Wind power by swept area at Wuennenberg by extrapolation methods from Luckau data

In literature, it has been proved that Weibull distribution cannot represent all wind regimes encountered in nature. Forexample, the distributions whose null speed percentage is higher or in the case of bimodal distributions [4]. Besides, in estimating the Weibull parameters, the uncertainty is inevitable, essentially when there is not enough historic data [18]. Some researchers have been conducted to ameliorate the precision of wind modeling by Weibull, adopting Neural Network approach. In the paper of Celik et al [13] a GFNN (Genetic Fuzzy Neural Network) model, with three variables at the input, such as Weibull function parameters c and k and average value of wind speed, has been developed to estimate the probability distributions of wind speed, annual energy production of wind, and capacity factor in a very windy region at Turkey. Results have proven that a GFNN enacts errors smaller than by Weibull function and higher coefficient correlation testing by nine years recorded data. Authors in reference [19] have found that neural approach allows determining more precisely characteristics of wind energy of Dakar site: wind speed frequencies, total period time of wind speed for one year, available energies; recoverable and produced for two different heights, in comparison with Weibull.

For applying the MCP method, as the name indicates, a close correlation values between reference and target short term samples is required. Or, it is not the case in complex terrains.

In this work, a Levenberg-Marquardt optimization based Back-propagation and Bayesian regularization based LMBP have been used to train Artificial Neural Network and estimate wind speed distributions. Given the fact that wind speed varies according to relief and altitude in atmospheric boundary layer, geographic characteristics; mean wind speed and standard deviation have been used in order to parameter the designated model. Out of the achieved results, we list the following remarks:

- Neural approach seems to be a promising alternative to replace some classical methods in wind assessment such as Weibull and MCP.

- Bayesian regularization based LMBP algorithm is less sensitive to the number of hidden neurons than Levenberg-Marquardt algorithm, but it converges slowly.
- Bayesian regularization based LMBP algorithm proves to be more appropriate and accurate for wind speed assessment against the LMBP algorithm.

4. Conclusions

This paper presents a preliminary study conducted to develop a model allowing the estimation of wind speed frequencies based on the Neural Network approach.

Distributions of wind speed have been estimated by different candidate architecture network. The sites were characterized by the monthly and standard deviation of wind speeds and geographical characteristics (longitude, latitude and altitude).

The results achieved by the proposed approach compared with Weibull and MCP methods results are closer to the measured data. The models are also used to assess the amount of power available in the wind speed, where the contributions of the different models were evaluated.

Results prove that learning by means of Bayesian regularization, in comparison with Levenberg–Marquardt learning algorithm, gives the best performance.

References

[1] P. Pinson and G. N. Kariniotakis, “Wind power forecasting using fuzzy neural networks enhanced with on-line prediction risk assessment,” in Power Tech Conference Proceedings, 2003 IEEE Bologna, 2003, vol. 2, p. 8 pp. Vol.2–.

[2] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. McKeogh, “Current methods and advances in forecasting of wind power generation,” Renew. Energy, vol. 37, no. 1, pp. 1–8, Jan. 2012.

[3] W.-Y. Chang, “An RBF Neural Network Combined with OLS Algorithm and Genetic Algorithm for Short-Term Wind Power Forecasting,” J. Appl. Math., vol. 2013, p. e971389, Mar. 2013.

[4] J. A. Carta, P. Ramírez, and S. Velázquez, “A review of wind speed probability distributions used in wind energy analysis: Case studies in the Canary Islands,” Renew. Sustain. Energy Rev., vol. 13, no. 5, pp. 933–955, juin 2009.

[5] S. S. Soman, H. Zareipour, O. Malik, and P. Mandal, “A review of wind power and wind speed forecasting methods with different time horizons,” in North American Power Symposium (NAPS), 2010, 2010, pp. 1–8.

[6] [G. Giebel, R. Brownsword, G. Kariniotakis, M. Denhard, and C. Draxl, “The state-of-the-art in short-term prediction of wind power: A literature overview,” ANEMOS. plus, 2011.

[7] K. G. Sheela and S. N. Deepa, “Review on methods to fix number of hidden neurons in neural networks,” Math. Probl. Eng., vol. 2013, 2013.

[8] M. T. Hagan and M. B. Menhaj, “Training feedforward networks with the Marquardt algorithm,” IEEE Trans. Neural Netw., vol. 5, no. 6, pp. 989–993, Nov. 1994.

[9] K. Matsuoka and J. Yi, “Backpropagation based on the logarithmic error function and elimination of local minima,” in 1991 IEEE International Joint Conference on Neural Networks, 1991, 1991, pp. 1117–1122 vol.2.

[10] F. D. Foresee and M. T. Hagan, “Gauss-Newton approximation to Bayesian learning,” 1997, vol. 3, pp. 1930–1935.

[11] D. J. C. MacKay, “Bayesian Interpolation,” Neural Comput., vol. 4, no. 3, pp. 415–447, May 1992.

[12] R. Kollu, S. R. Rayapudi, S. V. L. Narasimham, and K. M. Pakkurthi, “Mixture probability distribution functions to model wind speed distributions,” Int. J. Energy Environ. Eng., vol. 3, no. 1, p. 27, Oct. 2012.

[13] A. N. Celik and M. Kolhe, “Generalized feed-forward based method for wind energy prediction,” Appl. Energy, vol. 101, pp. 582–588, Jan. 2013.

[14] A. Redouane, D. Taoukil, A. El Bouardi, T. Ajzoul, and H. Ezbakhe, “Comparative Study of Weibull Parameters for Wind Speed Data of Northern Morocco.”

[15] J. Wen, Y. Zheng, and F. Donghan, “A review on reliability assessment for wind power,” Renew. Sustain. Energy Rev., vol. 13, no. 9, pp. 2485–2494, Dec. 2009.

[16] C. Nemes and F. Munteanu, “Reliability consideration on wind farms energy production,” in 2012 13th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM), 2012, pp. 183–187.

[17] Z. O. Olaofe and K. A. Folly, “Statistical Analysis of Wind Resources at Darling for Energy Production,” Int. J. Renew. Energy Res. IJRER, vol. 2, no. 2, pp. 250–261, 2012.

[18] S.-D. Kwon, “Uncertainty analysis of wind energy potential assessment,” Appl. Energy, vol. 87, no. 3, pp. 856–865, Mar. 2010.

[19] L. Thiaw, G. Sow, S. S. Fall, M. Kasse, E. Sylla, and S. Thioye, “A neural network based approach for wind resource and wind generators production assessment,” Appl. Energy, vol. 87, no. 5, pp. 1744–1748, May 2010.Bowman, M., Debray, S. K., and Peterson, L. L. Reasoning about naming systems. *ACM Trans. Program. Lang. Syst.* 15, 5, 795-825, 1993.

[20] Ding, W. and Marchionini, G. *A Study on Video Browsing Strategies*. Technical Report. University of Maryland at College Park, USA, 1997.

[21] Fröhlich, B. and Plate, J. The cubic mouse: a new device for three-dimensional input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Netherlands, April 2000.

[22] Tavel, P. *Modeling and Simulation Design*. AK Peters Ltd., Natick, MA, 2007.

[23] Sannella, M. J. *Constraint Satisfaction and Debugging for Interactive User Interfaces*. Doctoral Thesis, University of Washington, USA, 1994.

[24] Forman, G. An extensive empirical study of feature selection metrics for text classification. *J. Mach. Learn. Res.* 3, 1289-1305, March 2003.

[25] Brown, L. D., Hua, H., and Gao, C. A widget framework for augmented interaction in SCAPE. In *Proceedings of the 16th Annual ACM Symposium on*

User Interface Software and Technology, Vancouver, Canada, November 02 - 05, 2003.

[26] Yu, Y. T. and Lau, M. F. A comparison of MC/DC, MUMCUT and several other coverage criteria for logical decisions. *J. Syst. Softw*. 79, 5 , 577-590, May 2006.

[27] Spector, A. Z. Achieving application requirements. In *Distributed Systems*, S. Mullender, Ed. ACM Press Frontier Series. ACM, New York, NY, USA, 19-33, 1989.