# Short-term wind speed forecasting by combination of neural networks

Ramón Velo (ramon.velo@usc.es). Mª Dolores Fernández. Manuel R. Rodríguez.

Agroforestry Engineering Department. Polytechnic School.

University of Santiago de Compostela.

27002. Lugo. Spain.

## 1. Abstract

The aim of this work is to improve the prediction of wind speed in the next hours by the combined use of several artificial neural networks. The employment of Multilayer Perceptron networks (MLP) for a month period with ten-minute values of wind speed and direction and the prior choice of these inputs through classifiers networks such as Self-Organizing Maps (SOM) and Learning Vector Quantization (LVQ) is evaluated. The adjustment obtained in the prediction is compared with different methodologies.

## 2. Keyword

Self-Organizing Maps, Learning Vector Quantization, Multilayer Perceptron, Wind speed, Short-term wind speed forecasting.

## 3. Body

### 3.1. Introduction

Short-term predictions are usually based on time series analysis to complex structures as computational fluid dynamic, neural networks and fuzzy logic. Over recent years, in order to improve the prediction of the wind speed in the next few hours, it is using new artificial neural networks (ANN). One of the advantages of the ANNs is its ability to model the problems conditioned by multiple factors and complex relationships between variables to furnish nonlinear relationships between them [1].

Multi-model techniques can provide consensus predictions by linearly combining individual model predictions according to different weighting strategies. Different types of neural networks compete with statistical methods and hybrid models. Salcedo *et al.* [2] proposed an integrated approach with a mesoscale model and neural network system. The model integrates a global numerical weather prediction model and observations at different heights as initial and boundary conditions for the mesoscale model. The results of this model are processed using a neural network for forecasting wind speed. Different number of neurons in the hidden layer can be considered, and the results obtained will depend on this number. The values of MAE obtained are small, varying from values about 1.45 to values of 2.2 m/s at most.

Li *et al*. [3] proposed a two-step methodology for accurate wind speed forecasting based on Bayesian combination algorithm, and three neural network models, such as, adaptive linear element network, backpropagation network and radial basis function network. MAE values obtained are less than 1.07%.

Cadenas and Ribera [4] introduced several ANN configurations for later comparison by error measures, which guarantee the performance and accuracy of the models chosen. The simplest model of two layers, with two input neurons and one output neuron, was the best for the short term wind speed forecasting, with mean absolute error values of 3.99.

Torres et al. [5] used the ARMA model to predict the hourly averaged wind speed and compared it with the persistence model. They concluded that the ARMA model outperformed the persistence model. For forecasting horizon of 1 h, the persistence had less errors than ARMA model, while for forecasts 10 h in advance, the errors of ARMA model are between 12% and 20% smaller than of persistence model.

The use of support vector machine as neural network algorithm to wind speed prediction gives good adjustments [6].

Physical models use variables such as terrain, pressure and temperature to calculate the next wind speed [7]. Other times are used as the first stage to predict the wind, which is supplied as an auxiliary input for other statistical models [8]. Combinations of physical and statistical wind speed forecasting models are frequently used in wind speed prediction problems arising in wind farms management. Artificial neural networks can be used in these models as a final step to obtain accurate wind speed predictions [9].

The application of Kalman filtering as a postprocessing method in numerical predictions of wind speed leads to the elimination of any possible systematic errors, even and contributing further to the significant reduction of the operation time [10].

The multilayer perceptron network (MLP) is a feed forward widely employed, formed of information units called neurons which are connected to each other and arranged in layers. Each neuron outputs an activation function which is applied to the weighted sum of its inputs. The logistic sigmoid function is a differentiable nonlinear activation function that performs a smooth thresholding suitable for artificial neural networks. In supervised learning is presented to the network a set of patterns together with the target output. Their weights are adjusted iteratively until the output tends to be desired, using detailed information on the error committed at every step [11].

For a perceptron, the network’s outputs are given by equation (1).

$$y_i = \sum_j \omega_{ij}^2 \tanh \sum_l \left(\omega_{jl}^1 X_l + \theta_j^1\right) + \theta_i^2 \quad (1)$$

The function's parameters are given by the set of weights. $\Omega = \{\omega^1_{jk}, \omega^2_{ij}, \theta^1_j, \theta^2_i\}$ Where k runs over the networks input (1 and 2), j runs over the hidden layer (5 and 10) and i runs over the network's outputs (1). Therefore, $\omega^1_{jk}$ are the weights associated with the connections between the network's inputs and the hidden layer and $\theta^1_j$ the bias added to the input of the hidden layer's neurons. Similarly, $\omega^2_{jl}$ are the weights associated with the connections between the hidden layer and the network's outputs and $\theta^2_j$ the bias added to the input of the neurons of the hidden layer.

Self-Organizing Maps (SOM) are neural networks that present an unsupervised learning system but competitive, in which the output neurons compete amongst themselves to be activated, with the result that only one is activated at any one time. As a result neurons are obliged to organize, forming a two-dimensional map.

Learning Vector Quantization (LVQ) is an adaptive method for data classification based on training data with the desired class information. LVQ is a nearest neighbour pattern classifier based on competitive learning and is a supervised learning algorithm of ANN.

During training a pairs of sample points (x) and their class labels are shown to learn. The closest feature vector $v_w$ winning feature vector is updated according to one of the following equations, depending on if feature vector $v_W$ classifies sample point x correctly or not.

If $x$ and $v_w$ belong to the same class: $v_w(\mathrm{t}+1) = v_w(t) + \alpha(t)[x - v_w(t)]$

and if $x$ and $v_w$ belong to different class: $v_w(\mathrm{t}+1) = v_w(t) - \alpha(t)[x - v_w(t)]$

The term α(t) controls how large the movement of the feature vector should be in the update. The value of *α(t)* is updated after each training step according to equation (2).

$$\alpha(\mathrm{t}+1) = \frac{\alpha(\mathrm{t})}{1+\mathrm{p}\,.\ \alpha(\mathrm{t})} \quad (2)$$

where p=1 if sample point is classified correctly and p=-1 in another way. α(t) decrease when a sample is classified correctly and increase when an incorrect classification is made by the winning feature vector.

Training vectors are presented to the network as input and the Euclidean distance from the input vector to each of the feature vector is computed. Output neurons compete to be activated and only one gets it. The winning neuron is the one whose weight vector is closest to the input vector.

## 3.2. Methods

For the choice of input period is used primarily SOM neural network. They contain two layers, an input layer and output layer, so that the neurons are connected through adjustable weights or network parameters. The input layer consists of 2 neurons one for each input variable, 10-min values of wind speed and direction on the day before the target. They are chosen for this arrangement only the values of wind speed above 4 m/s. The neighboring function is Gaussian, learning rate is lineal *α(t)*=1/t and the maximum number of interactions that allowed was 500. The class with the largest number of values and always with a percentage above 40% is used as a criterion for the classification carried out by the LVQ network.

The LVQ networks are composed of three layers, an input layer, a competitive intermediate layer learns to classify input vectors into subclasses and the output layer transforms the competitive layer classes in target classifications. Training parameters were: 0.1 is chosen as α(0), 500 epochs, 2-20 of subclasses range, 120 s of maximum time.

The architecture of the MLP network was for all the simulations: 2 neurons in the input layer (10-min values of wind speed and direction), 5 or 10 neurons in the hidden layer and 1 neuron in the output layer (ten-min values of wind speed) (2_5_1 and 2_10_1). The utilization of MLP networks is divided into two phases, a training phase, during which a set of training patterns is used to determine the weights that define the neural model. For network training ten-minute values of wind speed and direction belonging to a period of 30 consecutive days is chosen.

The data selected correspond to stations located in Northwest Spain (Galicia): Ons (42,38º; -8,93º) at an altitude of 121 m and Punta Candelaria (43,71º; -8,05º) at an altitude of 254 m (Meteogalicia), both on the edge of the coast. Neural ToolboxTM software of MATLAB was used for the creation and resolution of the ANN models.

The performance of the models in forecasting accuracy was assessed using various evaluation criteria. Root Mean Square Error (RMSE) which is calculated according to the equation (3), Index of Agreement (IOA) in equation (4) and Mean Absolute Error (MAE) in equation (5).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - O_i)^2} \quad (3)$$

$$IOA = 1 - \frac{\sum_1^n (P_i - O_i)^2}{\sum_1^n (|P_i - O_{mean}| + |O_i - O_{mean}|)^2} \quad (4)$$

$$MAE = \frac{1}{n}\sum_{1=1}^{n}|P_i - O_i| \quad (5)$$

Where *P* is predicted output value and *O* is observed output value and n is number of data

The results obtained are compared with those provided by the persistent pattern and ARIMA (autoregressive integrated moving average) time series models more parsimonious.

## 3.3. Results

Table 1 shows the average values of the errors calculated at 50 simulations. It is notable the significant drop in RMSE (above 30 %) using the proposed method of electing the training period. In the prediction for next six hours, the error is less between 6 and 12% respect to 24 hours. We emphasize that the network 2_10_1 has the lowest error within neural networks analyzed, although the ARIMA model reaches a smaller RMSE.

| INPUT (ten-minute speed and direction values) | MODEL | RMSE | |
|---|---|---|---|
| | | 24 h | 6 h |
| 30 previous days to the target | PERSISTENCE | 3.36 | 2.51 |
| | ARIMA (2 0 0) | 2.67 | 1.74 |
| | ARIMA (2 1 0) | 3.19 | 2.23 |
| | 2_5_1 | 4.22 | 3.12 |
| | 2_10_1 | 4.40 | 3.02 |
| 30 days chosen by means LVQ and SOM (>40 %) | 2_5_1 | 2.83 | 2.49 |
| | 2_10_1 | 2.75 | 2.47 |

Table 1: RMSE average values for next hours

In Figure 1 the decrease in the RMSE obtained for the next 24 hours after applying the neural network 2_10_1, is observed. Whose horizontal axis represents the class size (%) that constitute the largest group obtained with LVQ network and subsequently used for training the MLP network.

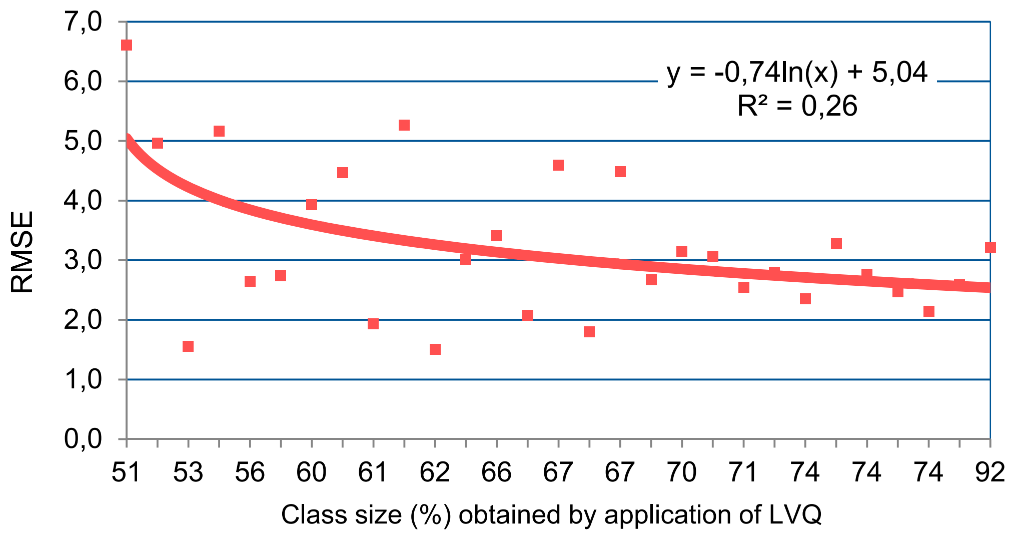


Figure 1: RMSE variation with chosen class size for LVQ neural network

A low coefficient of correlation is observed from sizes above 50% of the data that configure the LVQ class and that will allow to choose the input month for back propagation neural network. Other word is not decisive size selected once exceeds 50% class.

IOA values obtained for the method proposed against other analyzed options are observed in Figure 2. We note that only the ARIMA model (2 0 0) has better values in the IOA confirming the significant adjustment of the proposed method. The selection by the proposed method of the month is used as input, improves the fit in the forecast wind speed within 24 hours at the election the previous month.

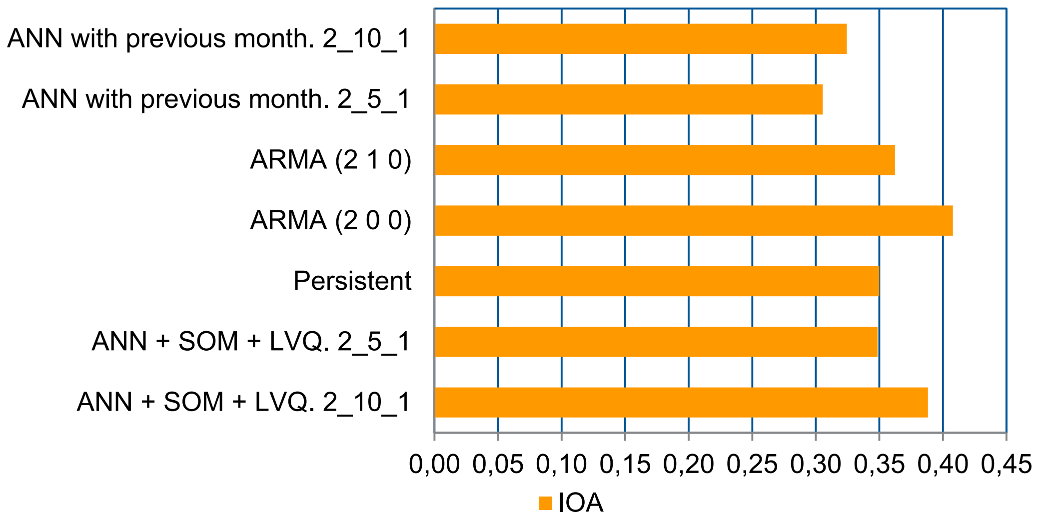


Figure 2: Index of Agreement values for next 24 hours

MAE values obtained for the method proposed against other analyzed options are seen in Figure 3.

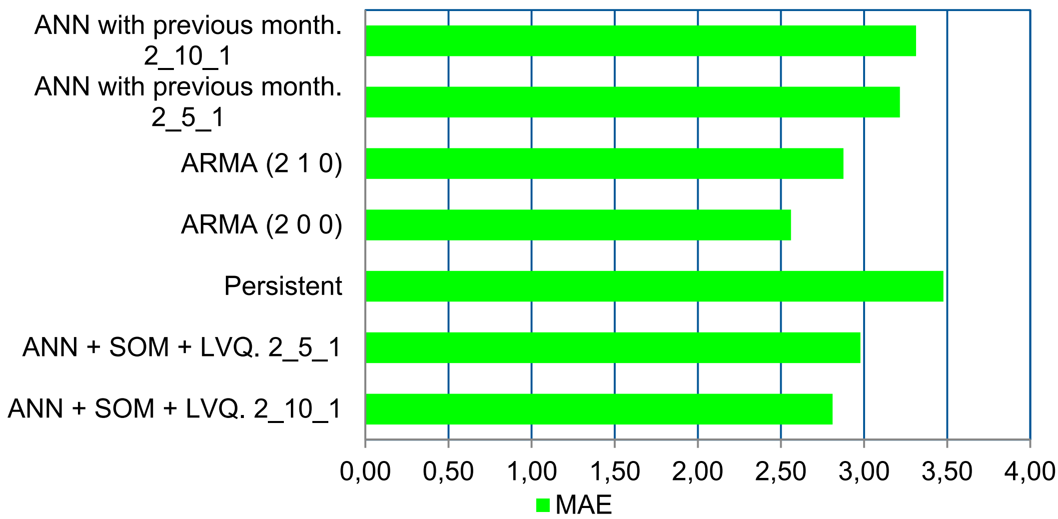


Figure 3: Mean Absolute Error values for next 24 hours

Only the mean absolute error is lower in the ARIMA model that the proposed method. The error is also reduced with respect to that obtained with neural networks using the previous month as input

## 4. Conclusions

In this work proposes a model to predict the wind speed in the coming hours. The results indicate that using networks LVQ and SOM for input selection improves prediction of the MLP neural networks.

The neural network 2_10_1, in combination with SOM and LVQ neural networks, provides better results in predicting the speed of the next 24 hours.

If the class size obtained by the application of the LVQ network contains more than 50 % of data, the value of RMSE is reduced. However does not grow significantly as the size of this class is incremented.

## 4. References

[1]. Ata R. “Artificial neural networks applications in wind energy”. *Renewable and Sustainable Energy Reviews* 2015; 49:534-562.

[2]. Salcedo S, Pérez AM, Ortiza EG, Portilla A, Prieto L, Paredes D. “Hybridizing the fifth generation mesoscale model with artificial neural networks for short-term wind speed prediction”. *Renewable Energy* 2009; 34:1451–1457.

[3]. Li G, Shi J, Zhou J. “Bayesian adaptive combination of short-term wind speed forecasts from neural network models”. *Renewable Energy* 2011; 36: 352–359.

[4]. Cadenas E, Rivera W. “Short term wind speed forecasting in La Venta, Oaxaca, México, using artificial neural networks”. *Renewable Energy* 2009; 34: 274–278.

[5]. Torres JL, García A, De Blas M, De Francisco A. Forecast of hourly average wind speed with ARMA models in Navarre. *Solar Energy* 2005; 79(1): 65–77.

[6]. Mohandes MA, Halawani TO, Rehman S, Hussain AA. Support vector machines for wind speed prediction. *Renewable Energy* 2004; 29: 939–947.

[7]. Lei M, Shiyan L, Chuanwen J, Hongling L, Yan Z. “A review on the forecasting of wind speed and generated power”. *Renewable and Sustainable Energy Reviews* 2009; 13: 915–920.

[8]. Hervás C, Salcedo S, Gutiérrez PA, Ortiz EG, Prieto L. "Evolutionary product unit neural networks for short-term wind speed forecasting in wind farms”. *Neural Computing and* Applications 2012; 21: 993–1005.

[9]. Sheela G, Deepa SN. “Neural network based hybrid computing model for wind speed prediction”. *Neurocomputing* 2013; 122:425-429.

[10]. Louka P, Galanis G, Siebert N, Kariniotakis G, Katsafados P, Pytharoulis I, Kallos G. “Improvements in wind speed forecasts for wind power prediction purposes using Kalman filtering”. *Journal Wind Engineering Industrial Aerodynamic* 2008; 96(12): 2348–2362.

[11]. Haykin S. *Neural networks a comprehensive foundation;* Second edition. Prentice-Hall: NJ USA, 1999.