Received: 4 June 2019 | Revised: 14 November 2019 | Accepted: 17 November 2019

DOI: 10.1002/we.2460

RESEARCH ARTICLE

WILEY

Wind power forecast using neural networks: Tuning with optimization techniques and error analysis

Gonçalo Nazaré[1] | Rui Castro[2] | Luís R.A. Gabriel Filho[3]

[1] IST—Instituto Superior Técnico, University of Lisbon, Lisbon, Portugal

[2] INESC-ID/IST, University of Lisbon, Lisbon, Portugal

[3] School of Sciences and Engineering, São Paulo State University (UNESP), São Paulo, Brazil

Correspondence
Rui Castro, INESC-ID/IST, University of Lisbon, Av. Rovisco Pais, 1049-001 Lisbon, Portugal.
Email: rcastro@tecnico.ulisboa.pt

Funding information
Fundação para a Ciência e a Tecnologia, Grant/Award Number: UID/CEC/50021/2019

Abstract

The increased integration of wind power into the power system implies many challenges to the network operators, mainly due to the hard to predict and variability of wind power generation. Thus, an accurate wind power forecast is imperative for systems operators, aiming at an efficient and economical wind power operation and integration into the power system. This work addresses the issue of forecasting short-term wind speed and wind power for 1 hour ahead, combining artificial neural networks (ANNs) with optimization techniques on real historical wind speed and wind power data. Levenberg-Marquardt (LM) and particle swarm optimization (PSO) are used as training algorithms to update the weights and bias of the ANN applied to wind speed predictions. The forecasting performance produced by the proposed models are compared with each other, as well as with the benchmark persistence model. Test results show higher performance for ANN-LM wind speed forecasting model, outperforming both ANN-PSO and persistence. The application of ANN-LM to wind power forecast revealed also a good performance, with an average improvement of 2.8% in relation to persistence. An innovative analysis of mean absolute percentage error (MAPE) behaviour in time and in typical days is finally offered in the paper.

KEYWORDS

artificial neural network, Levenberg-Marquardt, particle swarm optimization, short-term wind forecast

1 | INTRODUCTION

Wind power is the fastest growing source of renewable energy in the world. It represents a clean and sustainable source of energy and is in abundant supply, which helps to explain the growth in installed capacity of wind power plants in recent years. This implies the need to efficiently integrate the power generated from wind energy into existing power systems.

However, the increase in wind power penetration requires a number of issues to be addressed. Since the wind power has a cubic relationship with wind speed, any error in the wind speed forecast leads to a larger error in wind power production. This dependency in the stochastic nature of wind speed also causes uncertainty in wind power production, and unexpected variations of wind power output may increase the operating costs for the overall power system. Thus, the use of accurate short-term wind power forecast techniques is crucial in the planning of economical dispatch, aiming to an efficient and economical wind power integration and operation. This will enable to mitigate the undesirable effects of wind fluctuations in the operation of power systems, namely, by reducing the spin reserve margin capacity and increase wind power penetration.

The peer review history for this article is available at https://publons.com/publon/10.1002/we.2460.

Recently, with the development of artificial intelligence (AI), various AI methods for wind speed and wind power prediction have been developed and are being proposed, as for instance, artificial neural networks (ANNs), fuzzy logic and neuro-fuzzy, evolutionary algorithms (as particle swarm optimization [PSO]), and some hybrid methods. Also, wavelets and Markov chains are being used to capture the relevant patterns of the time series and act as preprocessing filters.

A key factor is the computation of the parameters or the weight coefficients of the model. This process would be enhanced if optimization techniques are used. One aim of this paper is assessing the use of optimization techniques to tune the parameters of the wind forecast models. The study will be focused on the use of ANN to predict wind speed, as this technique has proved to be a very promising one. Levenberg-Marquardt (LM) optimization and PSO performance in tuning the parameters of the network are to be compared.

An application to wind power forecast is also addressed in this paper using the ANN-LM structure. Another aspect that is dealt with in this paper is related to the forecasting errors behaviour. A common metrics of forecasting errors is the mean absolute percentage error (MAPE). This paper also aims at giving an innovative insight on how MAPE is correlated to the actual wind power, on how it changes over the course of 1 year, and on how it behaves on typical days.

Several methods are reported in the literature for wind speed and power forecasting. For instance, in Stefsos,[1] a comparison of various forecasting techniques is applied to mean hourly wind speed time series, such as linear autoregressive and autoregressive moving average (ARMA) models, feedforward and recurrent neural networks, adaptive neuro-fuzzy inference system (ANFIS), and neural logic networks (NLNs). The results showed that AI-based models outperformed the respective linear ones and all non-linear models exhibited, with a root mean square error (RMSE) lower than persistence model of 4.17% for the ANFIS model, 4.67% for ANN-LM, and 4.89% for NLN.

Kani and et al[2] proposed a very short-term wind speed model, employing ANN with Markov chains (ANN-MC). In their study, the short-term data patterns in wind speed are captured by ANN, and the long-term patterns are considered using Markov chain approach. The results showed that the proposed ANN-MC model reduces MAPE, as well as uncertainty.

In Bashir and El-Hawary,[3] a hybrid forecasting method combines PSO with ANN for short-term load forecasting. In this method, PSO is used to improve the performance of ANN, by adjusting ANN weights and bias in order to achieve a lower training error. PSO showed its ability to minimize the error function, and it was found that selecting proper values of network structure (input data and PSO parameters) is a major factor that affects the performance accuracy of the network.

Mandal et al[4] proposed a hybrid forecasting method combining wavelets, PSO, and ANN, considering real data of wind power, wind speed, wind direction, and temperature from wind farms located in the southern Alberta, Canada. This paper presents a short-term wind power forecasting methodology based on a data filtering technique using wavelets transform and a soft computing model based on ANN that is optimized by using PSO algorithm. In all test cases, the proposed model showed superior performance over the tested alternatives, demonstrating that the combining technique is more accurate, efficient, and capable of combining weather data with wind power data to improve short-term wind power forecast.

Catalão et al[5] developed a short-term wind power forecasting model combining neural networks and wavelet transform. The application of this model to short-term wind power in Portugal was effective, outperforming the persistence and neural network approaches.

Chang[6] described a wind power forecasting methodology based on backpropagation artificial neural networks (ANN-BP). To demonstrate the effectiveness of the proposed model, the method was tested on historical power generation data of a practical wind energy conversion system installed in Taichung coast. The developed model showed a good accuracy, with a mean absolute error of 2.6589%.

In another paper from Chang,[7] an enhanced particle swarm optimization (EPSO)–based hybrid forecasting method for short-term wind power forecast is proposed. The hybrid forecasting method combines the persistence model, ANN-BP, and the radial basis function (RBF) neural network. EPSO algorithm is used to optimize the weight and bias coefficients in the hybrid forecasting. The results demonstrate the effectiveness of the proposed hybrid method when compared with persistence, ANN-BP, and RBF, with an improvement in average MAPE by 1.45%, 10.04%, and 6.09%, respectively.

Several papers make a comparison between different computational intelligence techniques for wind prediction. For instance, Baptista et al[8] compare ANN, support vector machines, and ANFIS models. They conclude that the studied computational intelligence models perform better than the baseline persistence model, being ANFIS the model with the best results.

The work of Chang et al[9] presents an improved RBF neural network scheme with an error feedback scheme. The results are compared with conventional ANN-based forecasting methods, leading to an improved accuracy of the proposed model, while computational efficiency is maintained.

Advanced ANN techniques are also being applied to wind prediction. Khodayar et al[10] argue that ANN may fail to provide the accuracy that is required, because they apply shallow architectures with error-prone hand-engineered features. Therefore, they propose a deep neural network (DNN) architecture with stacked autoencoder and stacked denoising autoencoder for ultrashort-term and short-term wind speed forecasting.

The issue of selecting the appropriate forecasting method is approached in Madsen et al,[11] where a standardized and complete protocol, consisting of a set of criteria appropriate for the evaluation of a wind-power prediction system, is proposed.

In our paper, at first, ANN-LM forecasting methodology is applied to predict wind speed, the training set being obtained from the university automatic weather station. Other available weather parameters are studied in order to assess if they can improve wind speed prediction. Also, the

optimal training size is assessed. Then, PSO is combined with ANN to determine if weights and bias optimization by PSO can improve ANN-LM prediction performance.

Based on the conclusions of this study, the method that proved better is applied for grid injected wind power prediction, the training set using data kindly supplied by REN, the Portuguese transmission system operator (TSO). The effectiveness and efficiency of the proposed wind power forecasting methods is demonstrated by comparing the results with the widely used benchmark persistence method. Moreover, an innovative study on the MAPE characteristics is performed, namely, by assessing the correlation of the MAPE versus actual injected wind power, the MAPE time evolution, and the typical days MAPE.

The paper is organized as follows. In Section 2, the methodology used to achieve the proposed objectives is presented. The application of the methodologies to wind speed forecasting using weather data, as well as the obtained results and discussion, are exposed in Section 3. Section 4 deals with the application to injected wind power forecast, based on input injected wind power. Also, this section shows the results and discussion of the MAPE characteristics assessment. Finally, in Section 5, the main conclusions of the work are drawn.

2 | METHODOLOGY

This section presents two forecasting models to be compared, ANN and persistence models. Two techniques are presented in order to update ANN weight and bias, which are LM and PSO.

2.1 | Persistence model

The persistence method is the simplest among time series forecasting models. It is called the naive predictor because the prediction made is based on the last measured value. This method is very good for short predictions (from seconds up to few hours) but lacks accuracy on medium and large time-scales forecasts (from few hours up to days ahead). This method is usually used as a benchmark when compared with other forecasting models, so it will be also used in this investigation.

For a time series Y_t, given a historical set of data $H_t = \{Y_0, Y_1, Y_2, \dots, Y_t\}$, the forecast of the forthcoming value of Y_t $\widehat{Y}_{t+k|t}$, by a persistence process is given by Equation (1):

$$\widehat{Y}_{t+k|t} = Y_t. \tag{1}$$

2.2 | Artificial neural networks

ANNs are structures that enable to simulate human thinking and its capability to adapt to complex problems, as well as learning by experience. ANNs consist of a set of interconnected processing units, the neurons, which are able to produce outputs from inputs, each connection having an associated weight. The weights are adjusted through a supervised learning process, till the produced output corresponds to the desired output. To create an ANN, an historical dataset (observed inputs; desired output) is required. These samples are divided in three subsets: training, validation, and test.

The processing units are divided in layers. There may be several hidden layers, where the data processing is carried out. The inputs of each layer are the outputs of the previous layer, affected by a weight. Given a dataset with N samples, (p_i, a_i), where p_i is a vector of inputs and a_i is a vector of outputs, the ANN, with K hidden layers and activation function f, is given by

$$\psi_K(p_j) = \sum_{i=1}^{K} \beta_i f(w_i p_j + b_i),\ j = 1, \dots, N, \tag{2}$$

where w_i is the weight vector between the hidden layer i and the input layer, β_i is the weight vector between the hidden layer i and the output layer, b_i is a constant bias related to the hidden layer i, and $f(w_i p_j + b_i)$ is the output of the hidden layer i in relation to the input p_j.

During the training phase, all the weights and bias are adjusted so that an error surface is minimized, ie, the surface that is obtained by plotting the error between the observed and the desired outputs, for several possible combinations of weights and bias.

The most used error minimization algorithm is the LM one, whose objective function is defined by

$$C = \sum_{j=1}^{N} \left(\sum_{i=1}^{K} \beta_i f(w_i p_j + b_i) - a_i \right)^2. \tag{3}$$

The LM algorithm is an approximation to the Newton method. It was designed to approach second-order training speed without the need to compute the Hessian matrix. When the objective function has the form of a sum of squares, which is the case of ANN, the Hessian matrix is

approximated by (4), thus avoiding the computation of second-order partial derivatives.

$$\mathbf{H} = \mathbf{J}^{\mathbf{T}}\mathbf{J}, \tag{4}$$

where **H** is the Hessian matrix and **J** is the Jacobian matrix.

The LM algorithm consists in solving Equation (5):

$$\mathbf{J}^{\mathbf{T}}\mathbf{e} = \left[\mu\mathbf{I} + \mathbf{J}^{\mathbf{T}}\mathbf{J}\right]\boldsymbol{\delta}, \tag{5}$$

where **J** is the Jacobian matrix that contains first-order partial derivatives of a vector-valued function; in ANN case, the Jacobian is an N-by-W matrix, where N is the number of entries in the training set and W is the total number of parameters (weights and bias); **e** is the error vector containing the output error for each input vector used on training the network. **I** is the identity matrix; **δ** is the weight update vector to be found; μ is Levenberg's damping factor, which is adjusted at each iteration guiding the optimization process; if the error reduction is fast, a smaller value of μ can be used, bringing the algorithm closer to the Gauss-Newton algorithm, whereas if the error increases, it reaches the algorithm closer to the gradient descent direction.[12]

One of the most important tasks in developing a good time series forecasting model is the selection of the input variables, which determines the architecture of the ANN. Since there is not a systematic approach on choosing these input variables in AI-based models, statistical methods are employed to find relevant inputs, as suggested by Sfetsos.[1] These methods are the partial autocorrelation function (PACF) and the cross-correlation function (XCF) whose expressions are withdrawn from Box et al.[13]

The PACF refers to the correlation of a time series with its own past and future values:

$$r_k = \frac{\sum_{t=1}^{n-k}\left(Y_t - \overline{Y}\right)\left(Y_{t-k} - \overline{Y}\right)}{\frac{\sum_{t=1}^{n}\left(Y_t - \overline{Y}\right)^2}{n}}, \tag{6}$$

where r_k is the autocorrelation coefficient for lag k; n is the number of observations; Y_t is the observation at instant t; and $\overline{Y}$ is the average of total data.

On the other hand, the XCF is a standard method of estimating the degree to which two series are correlated. For data pairs (Y_{11}, Y_{21}), (Y_{12}, Y_{22}), … , and (Y_{1T}, Y_{2T}), an estimate of the lag k cross-correlation is given by

$$r_{Y_1Y_2(k)} = \frac{1}{T}\sum_{t=1}^{T-k}\frac{\left(Y_{1,t} - \overline{Y}_1\right)\left(Y_{2,t-k} - \overline{Y}_2\right)}{\sqrt{Var(Y_1)Var(Y_2)}}; \quad k = 0, 1, 2, \ \ldots, \tag{7}$$

where $r_{Y1Y2(k)}$ is the cross-correlation coefficient at lag k; $Y_{1,t}$ is the observation at instant t of data series Y_1; $Y_{2,t-k}$ is the observation at instant $t - k$ of data series Y_2; $\overline{Y}_1$ is the average of data series Y_1; $\overline{Y}_2$ is the average of data series Y_2; $Var(Y_1)$ is the variance of data series Y_1; and $Var(Y_2)$ is the variance of data series Y_2.

The data presented to the ANN are previously processed to fulfil the requirements of the network structure. After the definition of past observations that are relevant for future prediction, and selection of input variables through correlations methods, the data are organized into pairs of input vectors and desired outputs/targets vectors for training purposes.

2.3 | Particle swarm optimization

PSO algorithm is a group-based stochastic optimization technique for continuous non-linear functions. It is defined by the evolution of a population of particles, represented as vectors in an N-dimensional space. Each particle flies around the multidimensional search space with a velocity, which is continuously brought up to date by the particle's own experience and the experience of the particle's neighbours or the experience of the entire swarm.

The procedure to implement PSO algorithm can be defined in the following steps:

Step 1. Initiate particles positions, x, and velocities, v, in a D-dimensional problem, randomly and uniformly distributed across the design space.

Step 2. Update the velocities of all particles at iteration $k + 1$ using the particles fitness or objective values, which are functions of the particles current positions in the design space at iteration k. Knowing the fitness functions for all particles, the particles personal best fitness value, x_{Pbest}, can be identified and also the particle, which has the best global fitness in the entire swarm, x_{Gbest}. Then the velocity of each particle is updated:

$$v(k+1) = \omega v(k) + c_1 r_1 (x_{Pbest}(k) - x(k)) + c_2 r_2 (x_{Gbest}(k) - x(k)), \tag{8}$$

where $v(k)$ is the particle velocity at iteration k; $v(k + 1)$ is the particle velocity at iteration $k + 1$; $x(k)$ is the particle position at iteration k; $x(k + 1)$ is the particle position at iteration $k + 1$; c_1 is the cognitive acceleration coefficient; c_2 is the social acceleration coefficient; r_1 and r_2 are column vectors whose elements are independent pseudo-random numbers selected from a uniform distribution; and ω is a static inertia weight that provides preference for a particle to keep moving in the same direction it was following in the previous iteration.

Experimental works suggest that it is better to adjust the initial inertia to a high value and to adjust the final inertia to a smaller value.[14] With these criteria, the swarm has firstly a more global search exploration, and then, throughout the iterative process, the swarm search field becomes more accurate.

Step 3. Finally, the position of each particle is updated using the velocity vector:

$$x(k+1) = x(k) + v(k+1). \tag{9}$$

According to Mendes et al,[15] one can define two different topologies of PSO algorithm: PSO with local neighbourhood (*Lbest*) and PSO with global neighbourhood (*Gbest*). The main difference is how each particle moves towards the D-dimensional space. In *Gbest* PSO, each particle moves towards its best previous position and towards the best particle in the entire swarm; in *Lbest* PSO, each particle moves towards its best previous position and towards the best particle in its restricted neighbourhood. In other words, to apply *Lbest* PSO, in Equation (8), $x_{Gbest}(k)$ is replaced by $x_{Lbest}(k)$, which represents the particle that has the best local fitness in the neighbourhood. These two topologies will be employed, in order to find out how they can improve ANN forecasting accuracy.

3 | WIND SPEED PREDICTION

3.1 | Case definitions

The data available for wind speed prediction were provided by IST—University of Lisbon automatic weather station. The provided data contain mean hourly values of wind speed (WS), temperature (T), pressure (P), and humidity (H), for 2 years. The wind speed daily variation along one year is shown in Figure 1. The stochastic nature of wind speed is observed, with daily average wind speeds values changing from 1 to 10 m/s.

The main objective is to find how the additional weather parameters can help improving wind speed prediction. The combination of all four explanatory variables (measured quantities) leads to eight different cases that will be tested (cases definition is in Table 1).

3.2 | Performance metrics

In order to evaluate the performance of the forecasting models, a conventional accuracy measure is used: the MAPE, defined as,

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|W_i - \widehat{W}_i\right|}{W_i}100\%, \tag{10}$$

where W_i represents the actual wind speed at hour i, $\widehat{W}_i$ is the predicted wind speed for that hour, and N is the number of forecasted values.

MAPE measures the size of the error in percentage, which allows a better understanding of how good the proposed model is. MAPE has a known disadvantage when the actual values are too small as will be highlighted bellow.

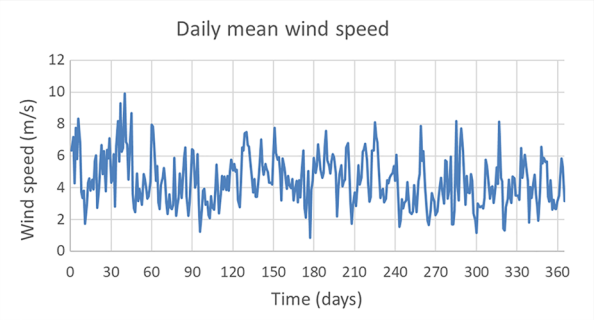


FIGURE 1 Daily mean wind speed at IST—University of Lisbon automatic weather station [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Case definitions: wind speed (WS), temperature (T), pressure (P), and humidity (H)

C#	1	2	3	4	5	6	7	8
Explanatory variable	WS	WS + T	WS + P	WS + H	WS + T + H	WS + P + H	WS + T + P	WS + T + P + H

3.3 | ANN-LM model

3.3.1 | Network architecture

PACF is used to find the linear autocorrelation coefficients from the wind speed data series in order to understand how past wind speed values contain relevant information for future wind speed predictions. On the other hand, XCF is employed to find if temperature, pressure, and humidity contain suitable information for future wind speed predictions. All correlation plots have a confidence interval of two standard deviations values, which correspond to approximate 95% confidence interval. Only coefficients higher than 0.6 are going to be chosen for input selection.

Figure 2 shows the PACF for the wind time series and the XCF for wind time series and temperature, humidity, and pressure time series, for the first year of available data.

It is possible to conclude that only lag 1 has a significant absolute value (higher than 0.6), which means that only one input is chosen for the ANN architecture, when forecasting wind speed using only wind speed past values. Temperature and humidity XCF show very small absolute coefficients values (lower than 0.3), which means that the variables temperature and humidity are most probably not linearly related to wind speed. Pressure XCF shows higher correlation values, which can be explained by the fact that wind speed is due to differences in atmospheric pressures. Yet the higher correlation value is lower than 0.4. For this reason, it was decided that the number of inputs to be presented to the

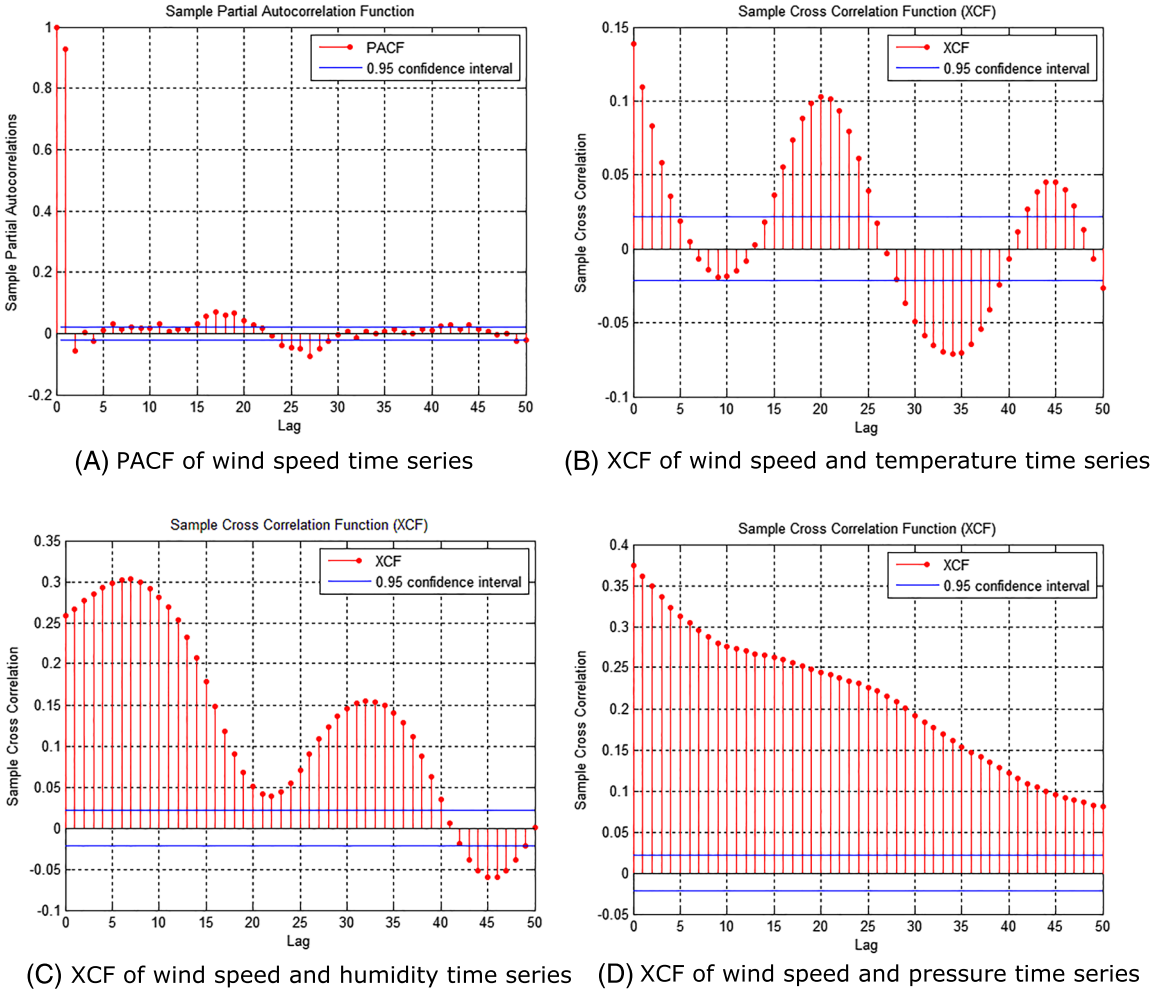


FIGURE 2 A, Partial autocorrelation function (PACF) of wind speed time series; B, cross-correlation function (XCF) of wind speed and temperature time series; C, XCF of wind speed and humidity time series; D, XCF of wind speed and pressure time series [Colour figure can be viewed at wileyonlinelibrary.com]

ANN in all XCF cases is also one. We note that the forecasting horizon is 15 minutes ahead and the input patterns dimensions depends on the number of explanatory variables shown in Table 1.

3.3.2 | Forecasting methodology

The number of past observations for ANN training can have a huge impact in prediction efficiency. If small amounts of past observations are considered in the training procedure, the ANN can become too specific relatively to that amount of data, which can prevent it to adapt to new situations outside the data range considered. On the other hand, if large values of past observations are considered, the ANN can become too generalized. In this work, seven different training sizes were tested in order to find the best number of past observations that are relevant for future prediction. The models considered are presented in Table 2.

In most of the available literature,[16-18] the number of neurons is calculated through try and error attempts. For setting the number of neurons in the hidden layer, the following method was used. First, a fixed number of neurons, equal to 5, is chosen in the hidden layer to evaluate the best model (training size) and the best case (weather parameters) in wind speed prediction. Then, after selecting the best forecast arrangement through the proposed accuracy measures, the number of neurons in the hidden layer will be optimized for this combination running the best forecast arrangement with the number of neurons changing from 1 up to 20.

3.3.3 | Results and discussion

In all simulations, the forecasting horizon considered is equal to 1 hour ahead repeated for 24 hours, which correspond to 24 values predicted. Following other papers strategy,[4,5,19,20] the results are separated into seasonality. The days chosen as representative of each season, and therefore chosen for prediction, were the 1st of February for winter, 1st of May for spring, 1st of August for summer, and 1st of November for autumn. The results obtained are compared with the persistence model.

Weather parameters and training size methodology for ANN model

From all the simulations performed for the seven training size models listed in Table 2, for the eight weather parameters cases (listed in Table 1), and for each representative day of the four seasons, it was concluded that models with short training size (models 4 to 7) have in general bad performance results, ie, worse than persistence model. On the other hand, large training size models (models 1 and 2) have also in general bad performance results and are again outperformed by persistence model.

For this reason, the strategy applied to find the best training size model was to find one model that can outperform persistence for all representative season days, with one particular weather parameters case (given in Table 1). This only happens with model 3 (2-wk training) for cases 2 (wind and temperature), 6 (wind, pressure, and humidity), 7 (wind, temperature, and pressure), and 8 (wind, temperature pressure, and humidity). For this reason, model 3 is chosen as training size model.

Figure 3 presents charts with MAPE and RMSE results for model 3 for all representative season days, in order to evaluate the best weather parameters case. All eight cases are presented in the figure just for a clean comparison since cases 2, 6, 7, and 8 were the best cases found.

The MAPE results in Figure 3A show that, in general, better MAPE results are achieved when additional weather parameters are used in wind speed prediction. The best MAPE result occurs in case 8 for February 1st (16.75%), on case 2 for May 1st (12.21%) and August 1st (9.33%), and on case 7 for October 1st (38.89%). As for the RMSE, Figure 3B indicates that the best RMSE take place on case 8 for February 1st (1.164 m/s), on case 2 for May 1st (0.757 m/s) and August 1st (0.560 m/s), and on case 4 for November 1st (0.860 m/s).

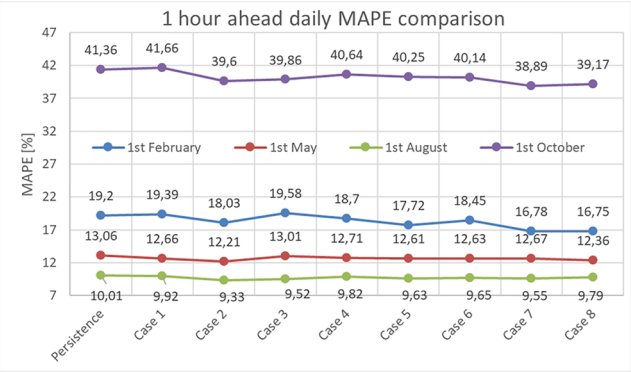
The conclusion is that there is not a clear combination of weather parameters that reveals the best results, and so it becomes more difficult to choose the best case. However, it can be seen that cases 2, 7, and 8 point to similar MAPE results. This allows to conclude that, besides temperature, the weather parameters humidity and pressure do not contribute significantly to a better wind speed forecasting for the data studied. As so, the best ANN model forecast chosen is given by case 2, which represents wind speed prediction through wind speed and temperature, along with model 3, which corresponds to 2-week training size.

Optimization of number of neurons in the hidden layer

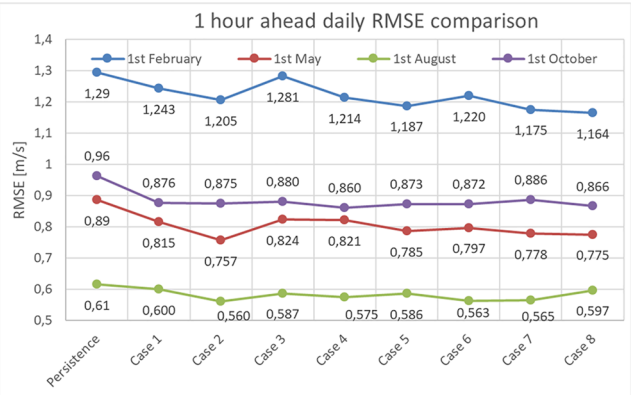
For an easier analysis, a different approach is made to evaluate the performance of the ANN models for each number of neurons in the hidden layer. A ratio is obtained by dividing the best ANN model MAPE result by the persistence model's MAPE, for each representative season day, and then an average value is calculated. Figure 4 shows the obtained results.

TABLE 2 Model definitions

M#	1	2	3	4	5	6	7
Training size	2 mo	1 mo	2 wk	1 wk	5 d	2 d	1 d



(A) 1 hour ahead daily MAPE comparison for model 3 (2 weeks training)
138x81mm (192 x 192 DPI)



(B) 1 hour ahead daily RMSE comparison for model 3 (2 weeks training)
136x85mm (192 x 192 DPI)

FIGURE 3 A, One hour ahead daily mean absolute percentage error (MAPE) comparison for model 3 (2-wk training); B, 1 h ahead daily root mean square error (RMSE) comparison for model 3 (2-wk training) [Colour figure can be viewed at wileyonlinelibrary.com]

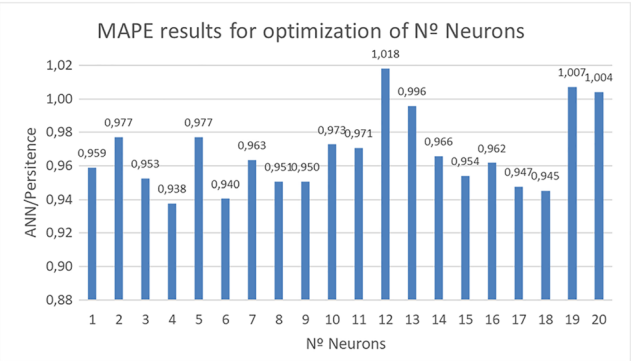


FIGURE 4 Ratio artificial neural network (ANN)/persistence daily mean absolute percentage error (MAPE) (representative season days average) for model 3 (2-wk training) and case 2 (wind and temperature) [Colour figure can be viewed at wileyonlinelibrary.com]

If the ratio, as presented in Figure 4 (model 3, case 2), is higher than 1, it means that persistence model leads to better results than the ANN model. On the other hand, if the results are lower than 1, it means that ANN model shows better results, in average throughout all representative season days.

The test results show that, with this ANN structure, the best number of neurons in the hidden layer is equal to 4 (lowest ratio: 0.938) when compared with persistence model. It is also evident that a wrong choice of the number of neurons can lead to very poor forecasts. For instance, when the number of neurons in the hidden layer is 12, 19, and 20, the ANN MAPE results are in average worse than persistence.

3.4 | ANN-PSO model

3.4.1 | Forecasting methodology

The purpose of PSO in ANN is to get the best set of weights (particles position) where several particles are trying to move to get the best solution. Two different topologies were applied (Local best and Global best referred in Section 2.3), in order to try to improve wind speed prediction for the best architecture found by ANN-LM model (model 3, case 2). In each topology, five different numbers of particles were tested (10, 20, 30, 40, and 50) with the following PSO network parameters presented in Table 3.

As mentioned before, PSO is used for the update of weights and bias in ANN training procedure as an alternative to LM optimization technique. The dimension of each particle has been calculated using two quantities: the weights and bias of the ANN architecture. In the final, the optimal weights and bias that minimize an error function are computed using PSO method.

Sometimes, particles tend to converge to a suboptimal solution before reducing the cost function to its true global minimum. Addressing this problem, the PSO algorithm was run three times, and the one with best performance was selected to perform wind speed forecast in each hour.

3.4.2 | Results and discussion

Simulations were performed for the number of particles mentioned before, for the topologies under assessment and for each representative day. The network structure was the one previously identified as the best one (2-wk training size and wind speed and temperature explanatory variables).

It was found that the MAPE results with the ANN-PSO algorithm are in general worse than the Best ANN-LM model found before and also worse than persistence model. Nevertheless, there are some situations, in which the ANN-PSO performs better than the ANN-LM model, but these are exceptions. Regarding CPU times, both ANN-LM best model and ANN-PSO for Global best and Local best topologies present very similar results.

Since a topology/number of particles that outperforms the best ANN-LM model could not be found, it can be concluded that the use of PSO technique to optimize weights and bias in ANN training is an ineffective technique as compared with LM backpropagation algorithm. Of course, this conclusion applies to the used time series.

ANN-LM performs slightly better than ANN-PSO. As so, ANN-LM will be used hereafter to perform wind power predictions.

4 | WIND POWER PREDICTION

In this section, the data available for wind power forecast were kindly provided by Redes Energéticas Nacionais (REN), the Portuguese TSO. The provided data contain records of injected wind power in the Portuguese power system, 15-minute averages for 5 years, from all wind farms in Portugal that have telemetry with REN.

Since the data available contain records of injected wind power in the Portuguese power system from of all wind farms in Portugal, the errors associated to forecasting wind power are expected to be lower when compared with wind speed forecast, dealt with in the previous section. This is foreseeable since when there are multiple wind farms connected to the power system, the wind power fluctuations from each wind farm represent a low change in total injected wind power, leading to a more smoothed wind power profile.

Figure 5 presents the daily injected wind power in the Portuguese power system in the last year of available data. It is observed that in the middle of the year, the injected wind power presents lower values, and for the early and later months, the opposite happens. This is expected

TABLE 3 PSO network parameters for wind speed forecast

Parameter	Value
Initial inertia weight ω_{max}	0.9
Final inertia weight ω_{max}	0.4
Static inertia weight ω	1.4
Maximum number of iterations $iter_{max}$	2000
Cognitive acceleration c_1	2
Social acceleration c_2	2
Search space range	(−1,1)

Abbreviation: PSO, particle swarm optimization.

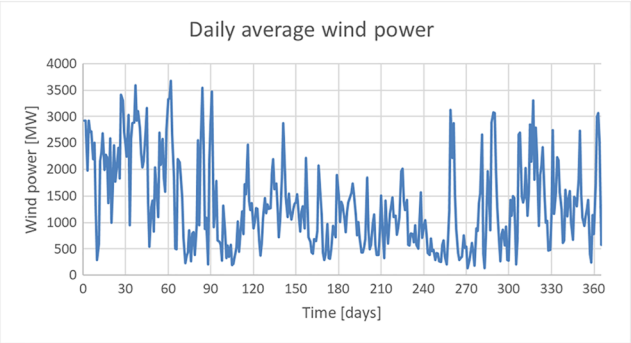


FIGURE 5 Daily average wind power [Colour figure can be viewed at wileyonlinelibrary.com]

knowing beforehand that in Portugal, spring and summer are normally seasons with low wind speeds and, on the other hand, autumn and winter present higher wind speeds.

In this section, ANN-LM structure is used. The same methodology that was used in the previous section to tune the ANN was applied to the wind power time series. The only difference is that no weather parameters are available and therefore the explanatory variable is wind power solely. Firstly, different training size intervals where tested for each representative day and the MAPE results compared with the persistence model. The obtained results allow to conclude that the best performer is the model with 1-week training size, achieving an average MAPE of 2.8% in the four representative days (average MAPE of persistence is 3.85%). As for the number of neurons, it was found that three neurons in the hidden layer is the best ANN structure, leading to a MAPE average improvement of 28% as compared with persistence.

It is to be noticed that a significant decrease in MAPE is observed in wind power forecast as compared with wind speed forecast. This is due to the aforementioned smoothing effect related to the great number of parks included in the real data, and as well to the fact that available data are now 15 minutes granular, instead of 1 hour for the wind speed case.

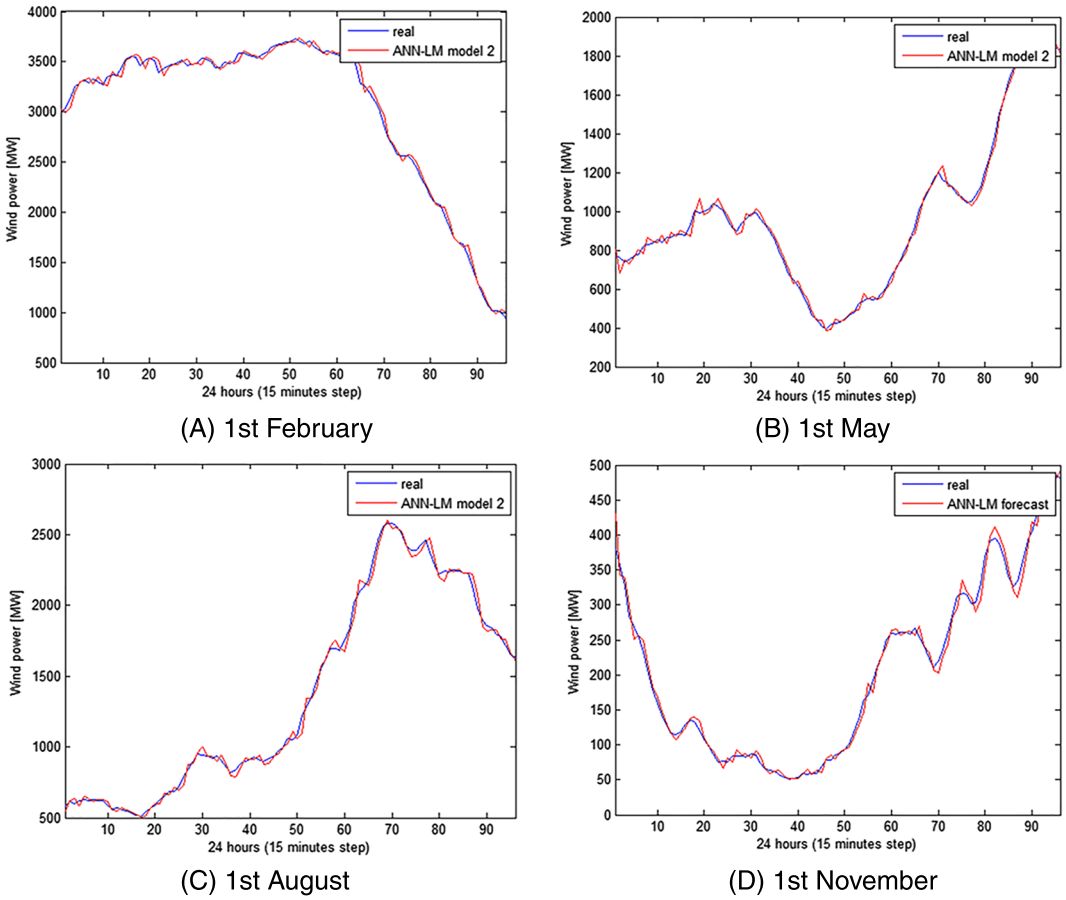


FIGURE 6 Actual wind power (blue) and best artificial neural network (ANN) topology (red) forecasting for (A) 1st of February, (B) 1st of May, (C) 1st of August, and (D) 1st of November [Colour figure can be viewed at wileyonlinelibrary.com]

Figure 6 shows the results of the comparison of best ANN-LM structure and actual wind power values for the four seasons typical days. The MAPE for each season representative days is as follows: 1st of February—1.45%; 1st of May—2.54%; 1st of August—2.46%; and 1st of November—4.67%. The average MAPE is 2.78%.

This ANN-LM structure is used to perform an analysis of the wind power forecast errors, as presented below.

4.1 | Correlation MAPE versus wind power

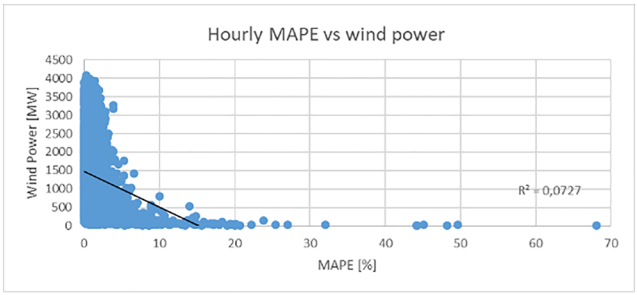
To assess the correlation between MAPE and wind power injections during 2014, scatter plots are made. Figure 7 shows the scatter plots for hourly (8760 values) and daily (365) granulations.

Figure 7 suggests that the MAPE increases when wind power decreases. This is expected because, for lower wind power values, one bad forecasted value represents a high percentage difference, corresponding to a higher MAPE index.

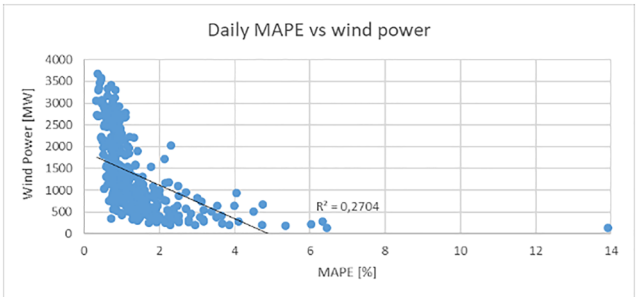
Hourly MAPE vs wind power (Figure 7A) shows an *R*-squared value of 0.073, which indicates that the hourly MAPE and actual wind power display very low correlation. However, in Figure 7B, the *R*-squared value increases to 0.27, therefore representing a higher correlation between the daily MAPE and the actual wind power. Lastly, Figure 7C shows the highest *R*-squared value (0.376), which evidences an even higher correlation between monthly MAPE and actual wind power.

4.2 | MAPE versus time

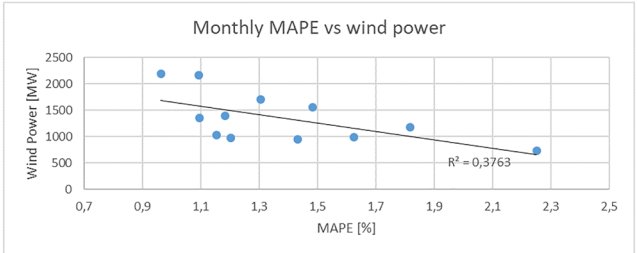
Figure 8 presents the time evolution of monthly and daily MAPE over the target year.



(A) hourly

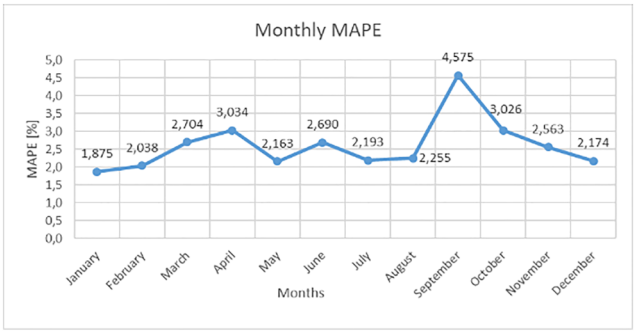


(B) daily

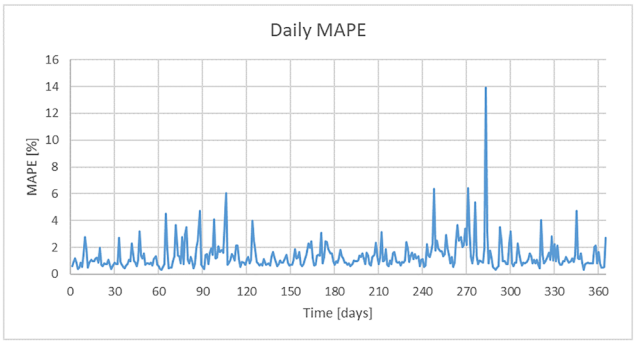


(C) monthly

FIGURE 7 Scatter plot for mean absolute percentage error (MAPE) versus actual wind power: (A) hourly, (B) daily, and (C) monthly [Colour figure can be viewed at wileyonlinelibrary.com]



(A) monthly



(B) daily

FIGURE 8 Mean absolute percentage error (MAPE) over the target year: (A) monthly and (B) daily [Colour figure can be viewed at wileyonlinelibrary.com]

Figure 8 (left) highlights the higher monthly MAPE results occurring in April, September, and October, with results over 3% error. For the other months, the monthly MAPE results are between 1.8% and 3%.

A more detail analysis is made in the same figure (right), showing the evolution of daily MAPE results over the year. The higher MAPE was obtained in October 10th (day 283) and the lower in October 17th (day 290). Analysing the records, one finds that the real injected wind power in October 10th was 6.5 MW (a very low value if one bears in mind that in this year, the installed capacity was 4500 MW); the forecasted value was 23.2 MW, which corresponds to the higher computed MAPE, but a difference of only 16.7 MW, ie, 0.4% of the installed capacity. As for October 17th, the real injected wind power was 3379 MW, and the forecast was 3406 MW, which corresponds to a difference of 27 MW, therefore higher than the October 10th one. This confirms that higher MAPE is expected for low wind power data and vice versa. To overcome this difficulty, Figure 9 is presented, in which absolute difference between actual and forecasted daily wind power is presented.

Figure 9 shows that the highest absolute difference between forecasted and actual wind power is always lower than 30 MW, ie, about 0.7% of the installed capacity (4500 MW). Also, it is possible to see that the highest absolute difference was on February 11th and the lowest in April 17th. Comparing Figure 9 with Figure 8 (right), the differences are apparent: The peaks and valleys do not occur in the same days. This allows to conclude that MAPE values do not match the absolute difference between actual and forecasted wind power, and so it depends strongly on actual wind power.

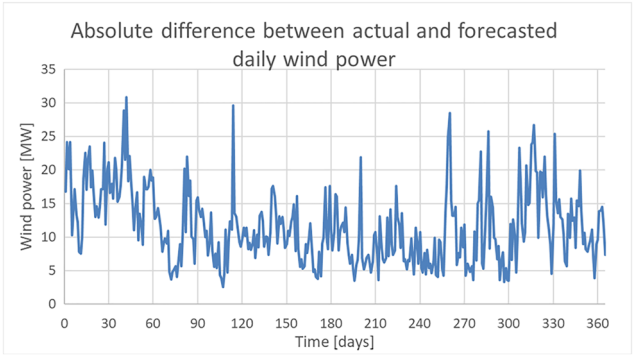


FIGURE 9 Absolute difference between actual and forecasted daily wind power [Colour figure can be viewed at wileyonlinelibrary.com]

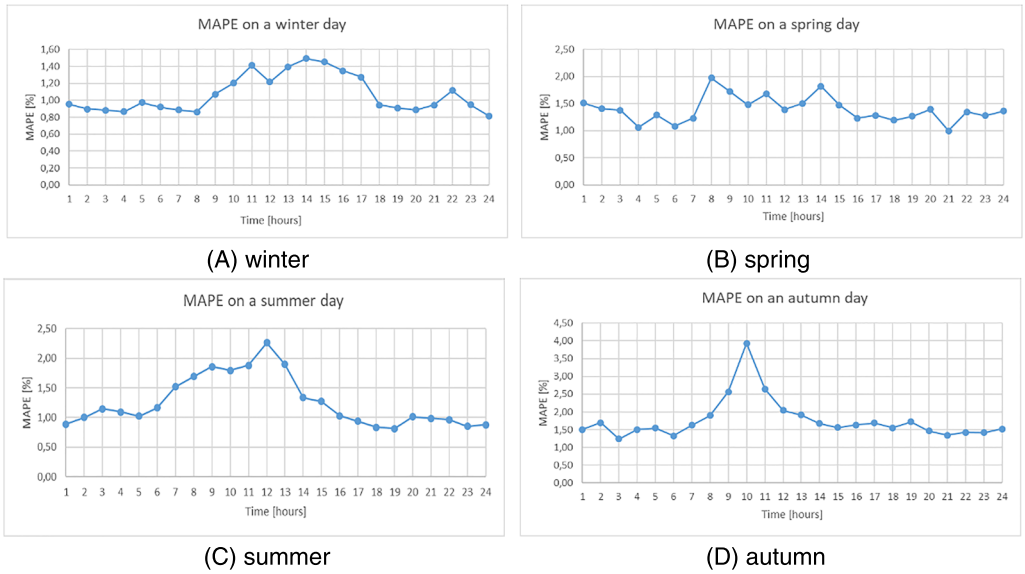


FIGURE 10 Mean absolute percentage error (MAPE) on typical days: (A) winter, (B) spring, (C) summer, and (D) autumn [Colour figure can be viewed at wileyonlinelibrary.com]

4.3 | Typical days MAPE analysis

Figure 10 presents the season typical days MAPE, for winter, spring, summer, and autumn.

From Figure 10, some conclusions may be drawn:

- Prediction errors are small, typically ranging in the [1%; 2%] interval.
- Prediction errors tend to be lower in the night than in the day.
- Maximum prediction errors occur by lunch time for summer and winter, and for autumn and spring, the maximum prediction error is expected to happen in the morning.
- The MAPE autumn pattern is very smooth, with errors around 1.5%, except a peak of 4% occurring in the morning.
- Minimum prediction errors are expected during the night for all typical days, the variation being between 1.0% and 1.5%.

5 | CONCLUSIONS

Two different forecasting models (ANN-LM and ANN-PSO) were used for wind speed prediction. Available data containing mean hourly values of wind speed, temperature, humidity, and pressure were used to train and validate the models.

The first conclusion to draw from ANN-LM implementation is that the selection of temperature and wind speed as input variables to present to the ANN is very important for an accurate wind speed forecast. On the other hand, adding the weather parameters pressure and humidity to wind speed forecast did not show any significant improvement.

Another important conclusion concerns the number of past observations that are relevant for future prediction. The strategy applied was to employ seven different training models, from 1-day to 2-month size. The results revealed that larger and shorter training size models had the worst error results and, on the contrary, medium-size models presented the best overall error results. Therefore, the best training model chosen was the one with 2-week size, since it presented better error results throughout all representative season days.

For the weather parameters (temperature and wind speed) and training model chosen (2-wk size), tuned in with ANN parameters, the ANN-LM outperformed persistence in all representative season days, with an average MAPE improvement of 5%.

The methodology of ANN-LM forecasting model was completed with the optimization of the number of neurons in the hidden layer for the best ANN-LM model found. The results showed that the ANN structure is very sensitive to changing the number of neurons in the hidden layer; thus, a proper selection is very important. The number of neurons that presented the best overall results was equal to 4, with an average MAPE improvement of 7.2%, throughout all representative season days.

Also, it was found that PSO technique used to optimize the weights and bias of the ANN was ineffective in both topologies studied (Global best and Local best). This conclusion is made because an ANN-PSO combination was not found that revealed better individual forecasted errors than ANN-LM and persistence, for all representative season days.

ANN-LM forecasting model was employed to wind power prediction, using 15-minute average records of injected wind power data in the Portuguese power system.

The first conclusion drawn was that choosing a proper number of past observations for ANN-LM training is important to achieve better forecasts results. Again, the shorter models tested revealed worst performances when compared with the middle-range ones. The training size model that revealed to be better was the one with 1-week training size, since it presented the best overall MAPE results throughout all representative season days, outperforming the persistence model.

Once again, analysing the number of neurons in the hidden layer revealed to be important in wind power prediction, where a proper choice of this number of neurons led to a better ANN-LM model. The number of neurons that presented the best overall results was equal to 3, with an average MAPE improvement of 28%.

With the best ANN-LM topology established, a wind power case-study application was offered correlating MAPE and actual wind power. It was concluded that MAPE indexes increase when wind power decreases, since it is expected that for lower wind power, high percentages errors are obtained.

Monthly MAPE analysis pointed to the conclusion that MAPE indexes ranges from 1.8% to 4.6% with April, September, and October presenting the highest error indexes. On the contrary, December, January, and February presented the lowest error indexes.

For the daily MAPE case, an important conclusion emerged: It is desirable to know the actual wind power when analysing MAPE indexes. The days with highest and lowest daily MAPE indexes were studied, and presented a difference, between actual and forecasted values, of 360% and 1% respectively. However, this corresponded to an absolute difference between actual and forecasted wind power values of only 16.7 MW for the highest daily MAPE index day and 27 MW for the lowest, which reinforces the conclusion drawn. These values represent 0.4% and 0.6%, respectively, of the installed capacity in this year.

Lastly, a typical day analysis was performed for each season that allowed to conclude that the prediction errors (MAPE) are normally smaller, between 1% and 2%, that prediction errors tend to be lower at night and higher during the day, and, for summer and winter, maximum predictions errors are expected for lunch time while, for autumn and spring, maximum prediction errors occurs in the morning.

As an overall conclusion, we would like to highlight that the conclusions drawn in the paper are specific to the case study (wind power injected in the Portuguese power system) and can be totally different for other case studies, because the analysis carried out is strongly dependent on the input data.

ACKNOWLEDEGMENTS

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2019.

ORCID

Rui Castro https://orcid.org/0000-0002-3108-8880
Luís R.A. Gabriel Filho https://orcid.org/0000-0002-7269-2806

REFERENCES

1. Sfetsos A. A comparison of various forecasting techniques applied to mean hourly wind speed time series. *Renew Energy*. 2000;21(1):23-35.
2. Kani SAP, Ardehali MM. Very short-term wind speed prediction: a new artificial neural network–Markov chain model. *Energ Conver Manage*. 2011;52:738-745.
3. Bashir, Z.A., El-Hawary, M.E. "Short-term load forecasting using artificial neural network based on particle swarm optimization algorithm". 2007 Canadian Conference on Electrical and Computer Engineering, 2007.
4. Mandal P, Zareipour H, Rosehart WD. Forecasting aggregated wind power production of multiple wind farms using hybrid wavelet-PSO-NNs. *Intl J Ener Res*. 2014;38(13):1654-1666.
5. Catalão JPS, Pousinho HMI, Mendes VMF. Short-term wind power forecasting in Portugal by neural networks and wavelet transform. *Renew Energy*. 2011;36(4):1245-1251.
6. Chang W-Y. Application of back propagation neural network for wind power generation forecasting. *Int J Digital Content Tech Appl*. 2013;7(4):502-509.
7. Chang W-Y. Short-term wind power forecasting using the enhanced particle swarm optimization-based hybrid method. *Energies*. 2013;6(9):4879-4896.
8. Baptista D, Carvalho JP, Morgado-Dias F. Comparing different solutions for forecasting the energy production of a wind farm. *Neural Comput Applic*. 2018;1-9.
9. Chang GW, Lu HJ, Chang YR, Lee YD. An improved neural network-based approach for short-term wind speed and power forecast. *Renew Energy*. 2017;105:301-311.

10. Khodayar M, Kaynak O, Khodayar ME. Rough deep neural architecture for short-term wind speed forecasting. *IEEE Trans Indust Infor*. 2017;13 (6):2770-2779.
11. Madsen H, Pinson P, Kariniotakis G, Nielsen, Henrik Aa. Nielsen, Torben S. A protocol for standardazing the performance evaluation of short term wind power prediction models. *Wind Eng*. 2005;29(6):475-489.
12. Demuth, Howard, Beale, Mark. "Neural network toolbox for use with MATLAB." (1993).
13. Box GE, Jenkins GM, Reinsel GC, Ljung GM. *Time series analysis: forecasting and control*. John Wiley & Sons; 2015.
14. Shi, Yuhui, Eberhart, Russell. "A modified particle swarm optimizer". 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence, 1998.
15. Mendes, R., Cortez, P., Rocha, M., Neves, J. "Particle swarms for feedforward neural network training". Proceedings of the 2002 International Joint Conference on Neural Networks IJCNN'02, 2002.
16. Bashir, Z., El-Hawary, M.E. "Short term load forecasting by using wavelet neural networks". 2000 Canadian Conference on Electrical and Computer Engineering. Conference Proceedings. Navigating to a New Era. 2000.
17. Catalão, J.P.S, Pousinho, H.M.I., Mendes, V.M.F. "Neural networks and wavelet transform for short-term electricity prices forecasting." 15th International Conference on Intelligent System Applications to Power Systems, (2009). ISAP'09, 2009.
18. Gomes P, Castro R. Wind speed and wind power forecasting using statistical models: autoregressive moving average (ARMA) and artificial neural networks (ANN). *Int J Sustai Energy Dev*. 2012;1(1/2):41-50.
19. Catalão JPS, Pousinho HMI, Mendes VMF. Hybrid wavelet-PSO-ANFIS approach for short-term wind power forecasting in Portugal. *IEEE Trans Sustain Ener*. 2011;2(1):50-59.
20. Haque AU, Mandal P, Kaye ME, Meng J, Chang L, Senjyu T. A new strategy for predicting short-term wind speed using soft computing models. *Renew Sustain Energy Rev*. 2012;16(7):4563-4573.

How to cite this article: Nazaré G, Castro R, Gabriel Filho LRA. Wind power forecast using neural networks: Tuning with optimization techniques and error analysis. *Wind Energy*. 2019;1–15. https://doi.org/10.1002/we.2460