

Local models-based regression trees for very short-term wind speed prediction

A. Troncoso, S. Salcedo-Sanz, C. Casanova-Mateo, J.C. Riquelme, L. Prieto

Keywords:

Wind speed prediction
Very short-term forecasting horizon
Regression trees

A B S T R A C T

This paper evaluates the performance of different types of Regression Trees (RTs) in a real problem of very short-term wind speed prediction from measuring data in wind farms. RT is a solidly established methodology that, contrary to other soft-computing approaches, has been under-explored in problems of wind speed prediction in wind farms. In this paper we comparatively evaluate eight different types of RTs algorithms, and we show that they are able to obtain excellent results in real problems of very short-term wind speed prediction, improving existing classical and soft-computing approaches such as multi-linear regression approaches, different types of neural networks and support vector regression algorithms in this problem. We also show that RTs have a very small computation time, that allows the retraining of the algorithms whenever new wind speed data are collected from the measuring towers.

1. Introduction

With the wind energy booming of the last decade, there has been an increasing interest of designers and practitioners in wind-engineering related problems, such as wind farm design, planning or management. Some of these problems arise from the analysis of wind measuring towers data, situated in wind farms or wind farms prospection areas. The data of these measuring towers can be used to evaluate prospective locations for new wind farm placement [1,2], to reconstruct wind speed series using two-sites correlation models [3], to estimate mean wind speed from short-term data [4], or to obtain short-term and long-term wind speed prediction within the wind farms, etc. In this paper we are interested in this latter problem, specifically in the problem of very short-term wind speed prediction from neighbor towers. The problem is important, since in different countries like USA hourly energy prediction is required in every wind farm installed. Very short-term wind (or power) prediction is also important to ensure stability of the whole electric system in cases of large penetration of wind energy.

Several previous works have tackled wind speed prediction problems in wind farms using data from measuring towers. The majority of previous approaches use modern regression techniques, many of them based on soft-computing algorithms such as neural networks: multi-layer perceptrons [5–11], fuzzy-based neural approaches [12], two-hidden layer neural networks [13], fast training neural approaches [3], Support Vector Machines [14], Abductive networks [15], Bayesian networks [16], probabilistic methods [17] or generalized mapping regression [18]. In spite of this huge work on modern methods for wind speed prediction from measuring towers, there are still some margin for improvement, coming from methodologies that have been under-explored in this problem. One of these methodologies is regression trees, which have not been, to our knowledge completely exploited in wind speed prediction problems.

This paper aims to evaluate the performance of several regression tree models in a problem of very short-term wind speed prediction from reference stations or measuring towers. Specifically, we evaluate eight different regression tree models, including both linear and non-linear algorithms. We compare the regression trees with several existing soft-computing approaches, such as neural networks (multi-layer perceptrons, Extreme Learning Machines), Support Vector Regression algorithms, Group Method of

Data Handling (GMDH) networks, multi-linear regression techniques or other regression trees well-known as Classification and Regression Trees (CART) [21] or Chi-squared Automatic Interaction Detection (CHAID) [48], in a real problem of very short-term (hourly) wind speed prediction in a wind farm in Spain. The results obtained show that the regression trees based on local models are an interesting methodology in this kind of wind speed prediction problems, able to improve the results of the alternative algorithms used for comparison.

The rest of the paper is structured as follows: next section describes the main characteristics of regression trees, discusses the application of these algorithms in other prediction problems and finally describes the proposed regression trees evaluated in this paper. Section 3 presents the experimental part of the paper, where we carry out a comparative evaluation of the regression trees in a real wind speed prediction problem, and we also compare the performance of the best regression trees models with that of alternative soft-computing approaches for prediction. Section 4 closes the paper giving some final conclusions.

2. Proposed methodology: regression trees

Decision trees based on If-Then rules are one of the most popular methods used in machine learning for classification, since they offer results that can be easily interpreted. Thus, this approach obtains ordered sets of If-Then rules for prediction that produces understandable models [19].

In the last years, learning algorithms generating decision trees for classification problems have been extended by Kenesei and Abonyi [20] for predicting values of attributes when they are numeric. These extensions have led to regression trees. A regression tree is a decision tree in which the leaf nodes have been set as regression models, and therefore, continuous numeric values can be predicted. The trees are built by splitting the data based on the values of predictive attributes. Once the tree has been constructed, a regression model is computed for each node. Then the tree is pruned from the leaves while the estimated error decreases. The error for each node is the mean of the absolute difference between the predicted and the actual value of each instance of the training set that reaches the node. Usually, this mean is multiplied by a weight that takes into account the number of instances that reach the node. The process is repeated until all instances are covered by one or more rules.

The most popular method that builds RT is CART [21] in which the value of prediction for each leaf is the mean of all instances of the training set reaching that leaf. Quinlan [22] proposed the algorithm $M5'$, a CART's extension that builds trees with linear models in leaves, that is, piecewise linear trees. Moreover, this algorithm determines the split attributes by maximizing the reduction of variance instead of the information gain typically used to generate decision trees. The same learning algorithm used by the algorithm $M5'$ to generate regression trees was used by Torgo [23] with the purposes of building regression trees with more complex functional models in their leaves. Namely, these functional models were k -nearest neighbors models and kernel regressors.

The success and popularity of regression trees has motivated researchers to develop new learning algorithms to infer trees. For example, Kordos et al. [24] presents an evolutionary approach where a set of regression trees evolves to obtain the best tree. The mutation consists in shifting the split attributes and the crossover in exchanging particular tree subnodes between different trees. Sela and Simonoff [25] presented an algorithm called RE-EM trees to predict the price in online transactions. This method combines mixed effects models for longitudinal and clustered data with

learning algorithms of trees. Recently, an incremental learning algorithm to built trees in a data stream context has been published by Fidalgo-Merino and Nuñez [26]. On the other hand, Young et al. [27] and Toth and Eltinge [28] try to focus on the design of samples used in the training to infer the tree instead of the own learning algorithm. Nowadays, hybrids methods have turned into new tools to be explored such as the combination of feature selection methods with regression trees [29], the use of a regression tree to select the input data of a neural network [30] or ensemble of rules obtained by regression trees [31].

In addition, regression trees have been widely used to solve regression problems in domains as diverse as engineering, medicine or environment [32–34]. For example, in Leclere et al. [35] a combination of trees obtained by CART has been applied to model the occurrence of nine species from the Seine river in order to predict the presence or absence of them. In particular, the tree-based model was compared with a linear model and obtained the best results for eight species.

Regarding air quality time series, CART has been used by Choi et al. [36] in order to determine relationships between meteorological variables and pollutants. Namely, the years from 2007 to 2009 in South Coast of California were used and the mean surface wind speed, geopotential heights at 925 mbar, the upper air north-south pressure gradient, the daily minimum temperature, relative humidity at 1000 mbar and vertical stability were determined as the split variables in the tree, and therefore, they had the most influence on the pollutant levels. On the other hand, the concentration of pollutants has also been forecasted by CART in Garner and Thompson [37]. In this case, hourly ozone time series was collected from year 2005–2011 in Baltimore and the probability of exceeding the standard ozone for a certain day was predicted. Thus, the user could choose the probability threshold from which an excess of ozone can be predicted depending on the required number of false alarms. The results showed a hit rate 7.5% better than that of the National Air Quality Forecast Capability with the same false alarm rate.

With regard to applications related with the wind, CART has been used to discover relationships between wind speed and several weather conditions such as the atmospheric pressure, temperature, solar radiation and humidity in Mori and Awata [38]. The simulation results reported the station pressure and sea-level pressure as the most important variables explaining the wind speed. It can be noticed that the input variables for forecasting the wind speed are meteorological variables while in our work the input variables are the wind measurements of neighbors towers. Moreover, regression trees have been also proposed for predicting the wind energy production. For example, CART has been applied by Clifton et al. [39] to obtain the power output of a wind turbine from wind speed, turbulence intensity and wind shear. The results provided by CART were two or three times better than those of a standard power curve method typically used in the industry. In Fugon et al. [40] different data mining techniques including regression trees are compared for short-term wind power forecasting in three real wind farms. Hourly power production and weather forecasts comprising a period of 18 months from July 2004 to December 2005 were considered. The results revealed that the performance of regression trees was better than that of the remaining models.

Summarizing, most of applications use only the well-known CART algorithm to model or to obtain forecasts with regression trees. Despite the success of regression trees based on CART in a great and diverse number of applications, other models as piecewise linear tree or regression trees with the models in the leaves published by Torgo [23] have not received enough attention from the research community. Moreover, to the best of our knowledge,

these models have not been applied to the wind speed prediction when reference towers are used.

In this work, eight kind of novel regression trees structures have been applied to very short-term wind speed prediction when using reference towers. The difference between them is the different regression models considered in the leaf nodes. Specifically, we consider the following models:

Model 1 (M1): This model is model is a constant linear model. This constant value is the mean of the target variable for all instances of the training set that reach the leaf.

Model 2 (M2): This model is also a constant linear model, but the constant value is the median instead of the mean.

Model 3 (M3): Linear model in which the coefficients are computed by minimizing the mean square error.

Model 4 (M4): Linear model in which the coefficients are computed based on a minimum description length principle.

Model 5 (M5): This model is a reduced linear model. Once the linear model is obtained by the least square method, it is simplified as in M5' algorithm proposed by Quinlan [22]. In particular, a greedy search is used to determine which variables of the linear function can be removed by minimizing the estimated error.

Model 6 (M6): This model is a local non-linear model, based on the k -Nearest Neighbors algorithm. In this case, the forecasting is divided into two steps clearly differentiated. In a first step, the k nearest neighbors are searched for between the instances of the training set that reach the leaf, and secondly, the mean of the target variable of the k nearest neighbors is computed. Thus, the previously computed mean is the prediction.

Model 7 (M7): This model is also a local non-linear model based on the weighted k -Nearest Neighbors. The prediction is the weighted mean of the target variable of the k nearest neighbors and the weights w_i are given by a Gaussian kernel [23,46]:

$$w_i = \exp\left(\frac{d(x_i, n(x_i))}{\sigma}\right) \quad (1)$$

where d is the Euclidean distance between a point x_i and its i -th nearest neighbor $n(x_i)$, σ is the width of the Gaussian kernel and $\exp(\cdot)$ is the exponential function.

Model 8 (M8): This model is a local linear model based on a locally weighted linear regression (LWLR), in which the coefficients are computed by minimizing the weighted mean square error and the weights are given by the Equation (1).

Summarizing, six linear models (five global and one local) and two non-linear local models are considered to construct regression trees. A global model is based on finding a function that depends on the input data to model the output. However, a local model does not use the input data to predict the output but only the points close to the point to forecast. In general, the linear models present some limitations, since when the models in leaves are too simple, some functions can not be approximated. On the other hand, global models have a lower computational cost than local models, since the latter have to be rebuilt for each point of the test set.

The package CORElearn [41] (available in software R) has been used to generate the different regression trees described above. Default values (10) for the number of neighbors in models M6, M7 and M8 and 2 for the σ parameter in model M7 and M8 have been considered.

Thus, this methodology can be summed up in the steps shown in Fig. 1.

A tree regression when using in the leaves a constant linear model (M1), a linear model (M5), a non-linear local model (M6) and a linear local model (M8) for a toy example is presented in Figs. 2–5, respectively. The toy example is composed of 3 variables, $V1$, $V2$ and $V3$, and the target variable $V4$. As it was expected, a model 10-NN or LWLR(10,2) has to be computed for each point of the set test when local models are used in the leaves of the trees. In the case of the model M6, the 10 nearest neighbors of the point to predict have to be computed, and in the case of the model M8, a linear regression is computed by minimizing the weighted mean square error for the 10 nearest neighbors of the point to predict. For both cases, the nearest neighbors are computed from the instances that reach the leaf.

3. Experimental results

In this section we present the experimental part of the article, where we evaluate the performance of the different regression trees proposed in a real problem of hourly wind speed prediction in wind farms. First, we will show the available data and the methodology used in the experiments. Then, we present the results obtained in the real wind speed prediction problem tackled: a comparison between the different evaluated regression trees is first shown, and a comparison with alternative state-of-the-art regressions is finally discussed.

3.1. Available data and methodology

The proposed regression trees have been applied to wind speed prediction of a wind farm in northern Guadalajara, Spain, where 8 measuring towers collect wind speed data. Fig. 6 shows the location of the wind farm and the situation of the measuring towers within the farm.

The available dataset for this study is composed of hourly wind speed collected in the measuring towers from November 2008 to November 2010. Thus, eight wind time series are available, in particular, one for each tower. Note that towers present failures in measurement from time to time, and we need complete data in order to obtain significant results in terms of error, so we only consider time series where all the towers have wind speed measurements. Thus, discontinuous wind time series are obtained: a total of 239 time series of different length are available for each tower. The dataset has been divided into 70% for the training set and 30% for the test set as this proportion is the most common for the evaluation of machine learning techniques. Thus, the first 175 time series are used for the training set and the remaining 64 for the test set. The training wind speed series include 3061 wind speed hourly samples corresponding to hourly wind speed from November of the year 2008 to September of the year 2010, whereas the test series include a total of 1091 wind speed hourly samples corresponding to hourly wind speed from October to November of the year 2010. The wind speed prediction at time t for the target tower is carried out using the values of wind time series at time $t - 1$ for each one of the eight towers. It should be noted that the wind time series are discontinuous due to the failures in the towers, that is, the time $t-1$ does not always mean the previous hour.

The hourly wind speed has been recorded to visually illustrate its behavior and evolution in Figs. 7 and 8.

Fig. 7 shows the hourly averages and standard deviations of wind speed for the eight towers for 168 h of October 2010 in meter per second (m/s). It can be noted that, as expected, larger deviations occur during the peak hours, that is, hours corresponding to high wind speed values.

Input: Training and test set for each tower
Output: Errors for each tower

Load CORElearn package
for each tower T
 Read training set for the tower T
 Obtain model M using CORElearn
 Read the test set for the tower T
 Predict the test set by using the model M
 Evaluate errors by Eq. (2), Eq. (3), Eq. (4) and Eq. (5)
end for
return errors for each tower

Fig. 1. A general scheme of the methodology.

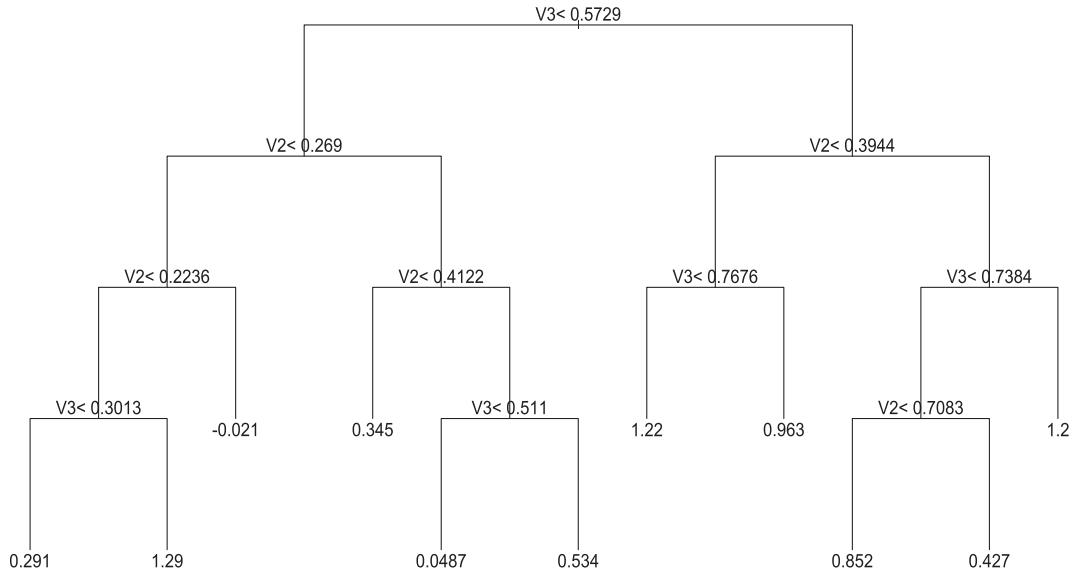


Fig. 2. A regression tree with M1 model in the leaves for a toy example.

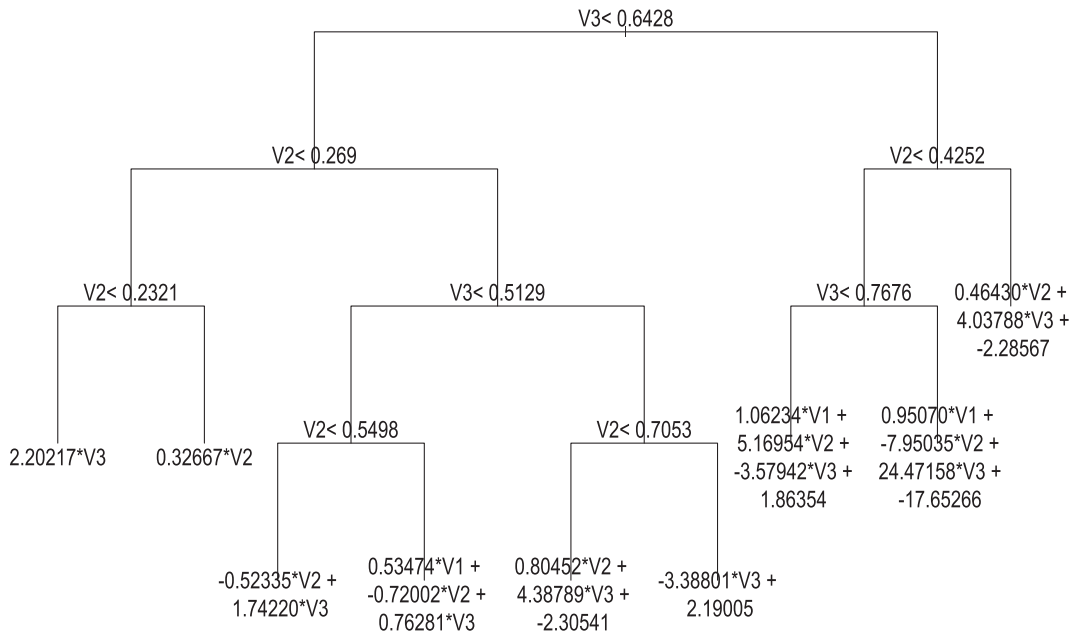


Fig. 3. A regression tree with M5 model in the leaves for a toy example.

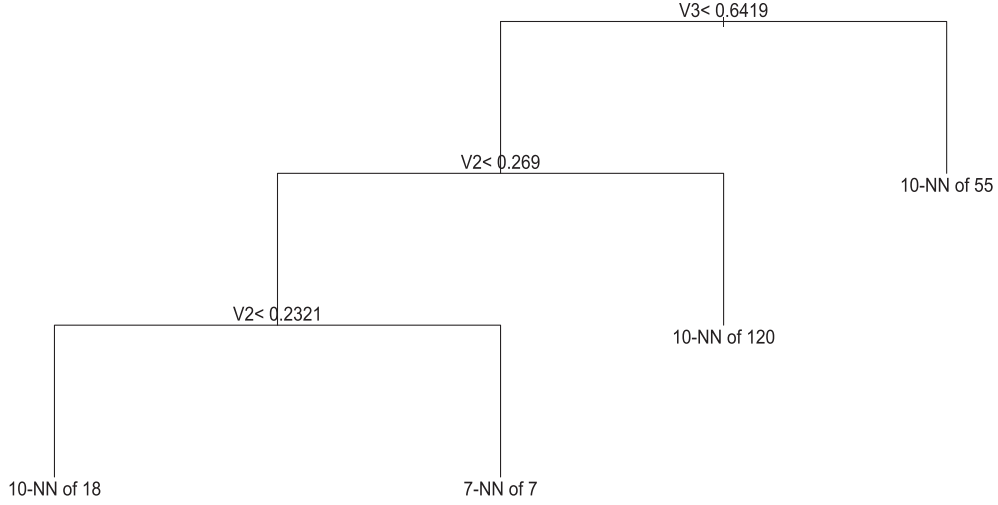


Fig. 4. A regression tree with M6 model in the leaves for a toy example.

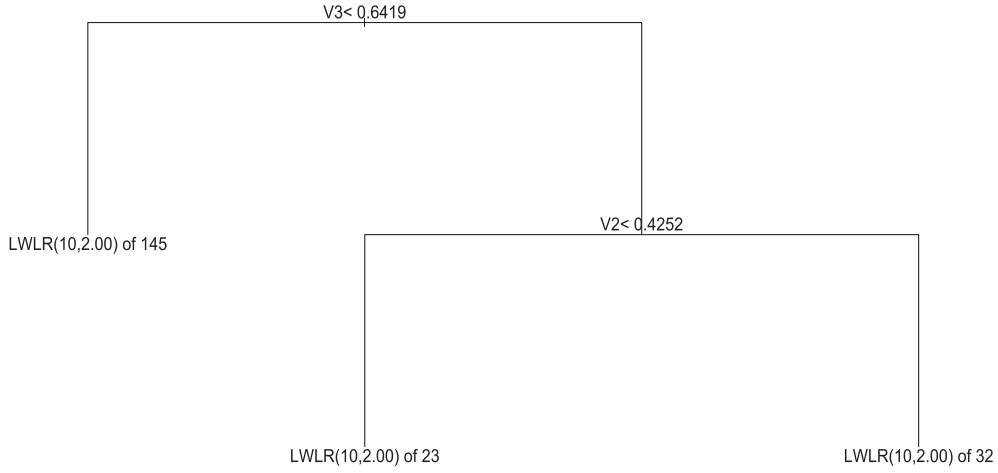


Fig. 5. A regression tree with M8 model in the leaves for a toy example.

The way wind speed time series is summarized in the histogram of Fig. 8, corresponding to Tower 1. It can be observed that the wind speed time series approximately follows a normal distribution. Moreover, a majority of wind speed points lay between 6 and 12 m/s.

In order to assess the performance of the regression trees, several quality measures [42,43] have been considered:

- Square Root of Mean Squared Error (MSE)

$$MSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{x}_i - x_i)^2} \quad (2)$$

where \hat{x}_i and x_i are the predicted and current wind speed at time i , respectively and N is the number of predicted values.

- Relative Mean Squared Error (RMSE)

$$RMSE = \frac{\sum_{i=1}^N (\hat{x}_i - x_i)^2}{\sum_{i=1}^N (\mu(\hat{x}_i) - x_i)^2} \quad (3)$$

where $\mu(\hat{x}_i)$ is the mean of the values of the variables \hat{x}_i .

- Mean Absolute Error (MAE)

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{x}_i - x_i| \quad (4)$$

- Relative Mean Absolute Error (RMAE)

$$RMAE = \frac{\sum_{i=1}^N |\hat{x}_i - x_i|}{\sum_{i=1}^N |\mu(\hat{x}_i) - x_i|} \quad (5)$$

- Coefficient of determination (R^2)

$$R^2 = \frac{1}{N \sigma_{x_i} \sigma_{\hat{x}_i}} \sum_{i=1}^N (x_i - \mu(x_i)) (\hat{x}_i - \mu(\hat{x}_i)) \quad (6)$$

where $\mu(x_i)$ is the mean of the values of the variables x_i and σ_{x_i} and $\sigma_{\hat{x}_i}$ are the standard deviations of x_i and \hat{x}_i , respectively.

- Index of Agreement (IA)

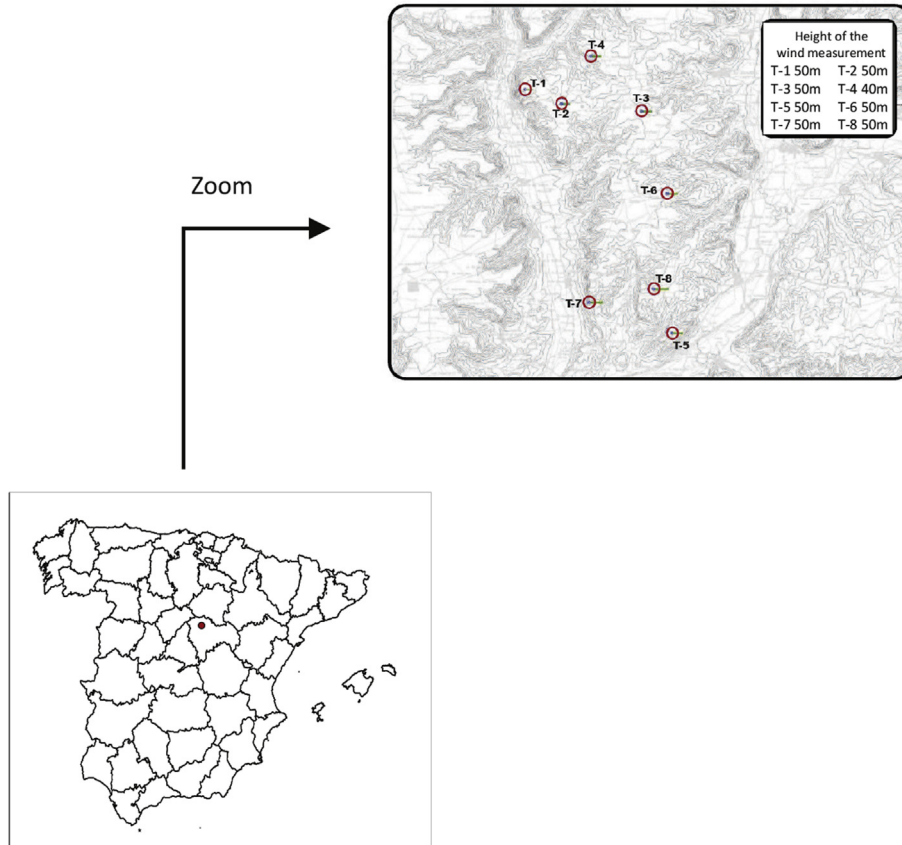


Fig. 6. Situation of the wind measuring towers in Spain and within the wind farm.

$$IA = 1 - \frac{\sum_{i=1}^N (\hat{x}_i - x_i)^2}{\sum_{i=1}^N (|\hat{x}_i - \mu(x_i)| + |x_i - \mu(x_i)|)^2} \quad (7)$$

3.2. Results and discussion

First, we carry out a comparison of the different regression trees proposed in this paper, in order to show major performance

differences among them. Tables 1–6 show the MSE, RMSE, MAE, RMAE, R^2 and IA errors obtained when applying the different proposed regression trees to the test set composed of 64 wind speed time series described in previous subsection. It can be appreciated that the best regression tree for most of the towers, in particular towers T1, T2, T4, T7 and T8, is usually any model based on nearest neighbors either M6 or M7.

Figs. 9 and 10 present the best and worst prediction for the tower eight for a week of the test set. The best and worst prediction is obtained when a regression tree with M7 and M8 model in the leaves is applied, respectively.

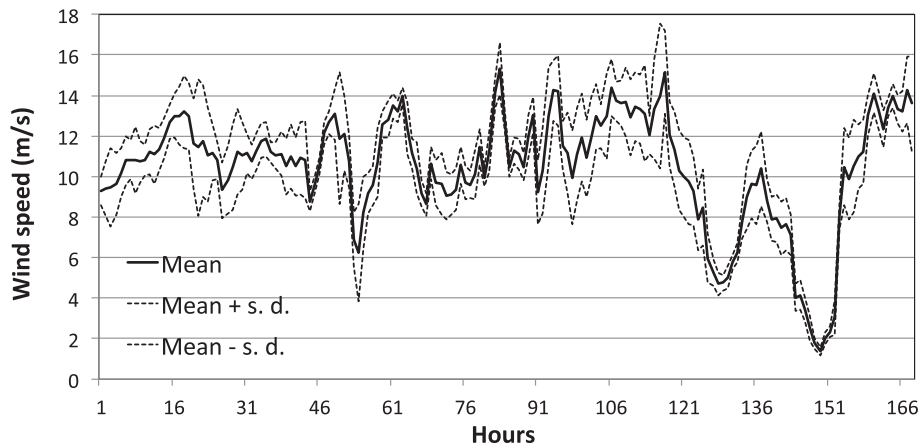


Fig. 7. Hourly average of the wind speed time series for the eight towers.

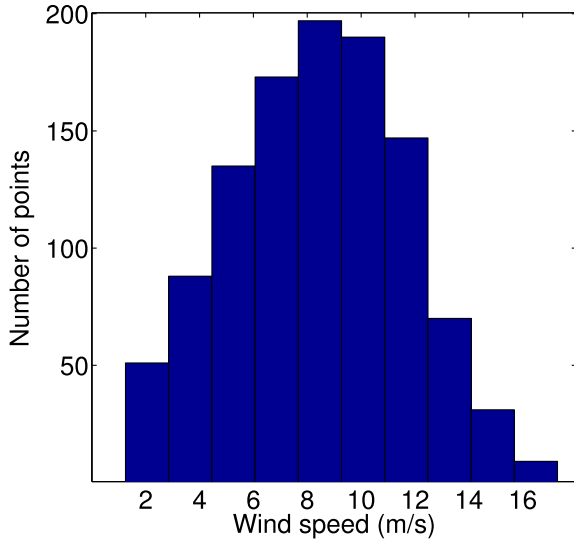


Fig. 8. Histogram of wind speed for Tower 1.

Table 1
MSE (in m/s) obtained in the wind speed prediction for each objective tower (TX stands for Tower X) by the different regression trees proposed.

	M1	M2	M3	M4	M5	M6	M7	M8
T1	1.410	1.485	1.410	1.441	1.396	1.338	1.353	2.946
T2	1.311	1.321	1.337	1.309	1.268	1.200	1.200	2.575
T3	1.388	1.333	1.260	1.257	1.321	1.235	1.235	2.546
T4	1.367	1.417	1.346	1.353	1.336	1.282	1.283	2.744
T5	1.469	1.490	1.403	1.469	1.415	1.385	1.385	2.886
T6	1.310	1.297	1.241	1.199	1.192	1.245	1.230	2.694
T7	1.669	1.647	1.667	1.784	1.696	1.608	1.605	3.176
T8	1.441	1.425	1.519	1.371	1.357	1.317	1.316	2.734
Avg	1.421	1.427	1.398	1.398	1.373	1.326	1.326	2.788

The best model for each tower is in bold.

Table 2
RMSE obtained in the wind speed prediction for each objective tower (TX stands for Tower X) by the different regression trees proposed.

	M1	M2	M3	M4	M5	M6	M7	M8
T1	0.187	0.208	0.187	0.196	0.184	0.169	0.173	0.818
T2	0.179	0.182	0.187	0.179	0.168	0.150	0.150	0.693
T3	0.228	0.211	0.188	0.187	0.207	0.181	0.181	0.769
T4	0.207	0.222	0.201	0.203	0.197	0.182	0.182	0.833
T5	0.181	0.186	0.165	0.181	0.168	0.161	0.161	0.699
T6	0.194	0.190	0.174	0.162	0.160	0.175	0.171	0.820
T7	0.201	0.195	0.200	0.229	0.207	0.186	0.185	0.726
T8	0.174	0.170	0.194	0.158	0.155	0.145	0.145	0.627
Avg	0.194	0.195611	0.187	0.187	0.181	0.169	0.169	0.748

The best model for each tower is in bold.

Table 3
MAE (in m/s) obtained in the wind speed prediction for each objective tower (TX stands for Tower X) by the different regression trees proposed.

	M1	M2	M3	M4	M5	M6	M7	M8
T1	1.085	1.148	1.054	1.083	1.055	1.017	1.019	2.327
T2	0.999	1.006	0.959	0.950	0.945	0.898	0.898	2.013
T3	1.065	1.020	0.922	0.929	0.968	0.927	0.927	1.965
T4	1.048	1.082	1.011	1.003	0.996	0.964	0.964	2.092
T5	1.133	1.134	1.048	1.074	1.048	1.054	1.054	2.244
T6	1.000	0.983	0.919	0.889	0.889	0.931	0.922	2.052
T7	1.256	1.229	1.214	1.297	1.243	1.204	1.204	2.473
T8	1.100	1.086	1.107	1.015	0.997	0.982	0.982	2.122
Avg	1.086	1.086	1.029	1.030	1.017	0.997	0.996	2.161

The best model for each tower is in bold.

Table 4
RMAE obtained in the wind speed prediction for each objective tower (TX stands for Tower X) by the different regression trees proposed.

	M1	M2	M3	M4	M5	M6	M7	M8
T1	0.409	0.433	0.397	0.408	0.398	0.383	0.384	0.878
T2	0.393	0.396	0.378	0.374	0.372	0.354	0.354	0.793
T3	0.448	0.429	0.388	0.391	0.407	0.390	0.390	0.826
T4	0.430	0.444	0.415	0.412	0.409	0.396	0.396	0.859
T5	0.398	0.399	0.368	0.378	0.368	0.371	0.371	0.789
T6	0.409	0.402	0.376	0.364	0.364	0.381	0.377	0.840
T7	0.408	0.399	0.394	0.421	0.403	0.391	0.391	0.803
T8	0.383	0.378	0.385	0.354	0.3471	0.342	0.342	0.739
Avg	0.410	0.410	0.388	0.388	0.384	0.376	0.376	0.816

The best model for each tower is in bold.

Table 5
 R^2 obtained in the wind speed prediction for each objective tower (TX stands for Tower X) by the different regression trees proposed.

	M1	M2	M3	M4	M5	M6	M7	M8
T1	0.794	0.794	0.789	0.810	0.823	0.824	0.823	0.443
T2	0.815	0.817	0.819	0.806	0.837	0.837	0.841	0.549
T3	0.772	0.781	0.784	0.806	0.793	0.810	0.810	0.493
T4	0.774	0.773	0.783	0.781	0.805	0.810	0.810	0.460
T5	0.804	0.808	0.829	0.812	0.835	0.826	0.831	0.539
T6	0.797	0.796	0.814	0.822	0.814	0.817	0.823	0.502
T7	0.790	0.785	0.780	0.782	0.791	0.809	0.809	0.463
T8	0.826	0.822	0.831	0.832	0.830	0.846	0.840	0.534
Avg	0.797	0.797	0.803	0.806	0.816	0.822	0.823	0.498

The best model for each tower is in bold.

Regarding the computational cost of the proposed regression trees in the wind speed prediction problem tackled, both learning (training) and forecasting computation times are shown in Table 7. The learning/training time is the time needed for generating the tree from the training set, whereas the forecasting time is the time consumed for computing the predictions from the test set. It can be observed that the time for building the tree ranges from 0.016 to 0.076 s depending on the model. Specifically, from 0.016 to 0.034 s for global models and from 0.055 to 0.076 s for local models, approximately. Note that the forecasting time is a very small fraction of the training time, as expected. Note that the training time provided by regression trees is very small, compared to the prediction time horizon (hourly prediction). This offers the possibility of retraining the trees almost in real time when new data are collected in the measuring towers.

In a second round of experiments, the regression trees that showed the best performance are compared versus state-of-the-art regression techniques. Specifically, Extreme Learning Machines (ELM) [44], GMDH networks (GMDH) [15], Multi-Layer perceptrons (MLP) [45], Support Vector Regression algorithms (SVMr) [46],

Table 6
IA obtained in the wind speed prediction for each objective tower (TX stands for Tower X) by the different regression trees proposed.

	M1	M2	M3	M4	M5	M6	M7	M8
T1	0.941	0.941	0.941	0.948	0.951	0.951	0.951	0.805
T2	0.947	0.948	0.950	0.947	0.955	0.955	0.956	0.854
T3	0.933	0.936	0.940	0.946	0.942	0.947	0.947	0.825
T4	0.934	0.935	0.939	0.939	0.945	0.946	0.946	0.812
T5	0.944	0.947	0.953	0.948	0.955	0.951	0.953	0.847
T6	0.943	0.942	0.949	0.951	0.948	0.949	0.951	0.825
T7	0.939	0.940	0.938	0.939	0.941	0.945	0.945	0.819
T8	0.951	0.951	0.954	0.954	0.954	0.958	0.956	0.840
Avg	0.942	0.942	0.945	0.946	0.949	0.950	0.951	0.828

The best model for each tower is in bold.

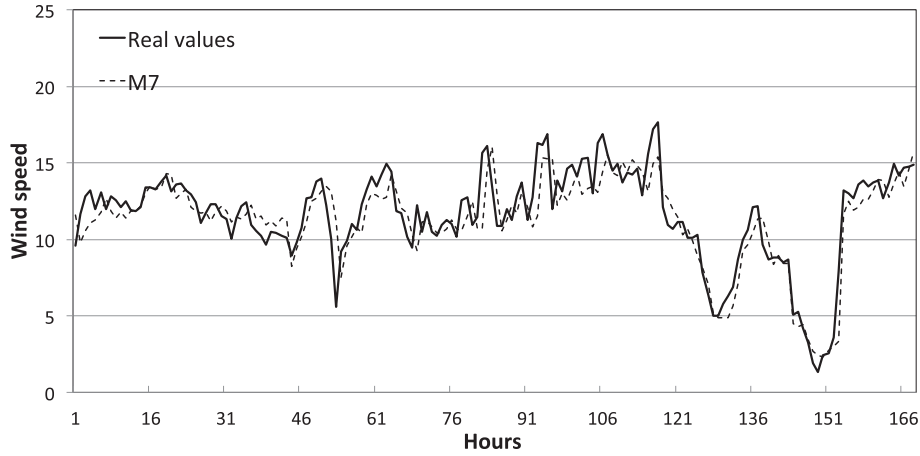


Fig. 9. Best prediction in a week of the test set.

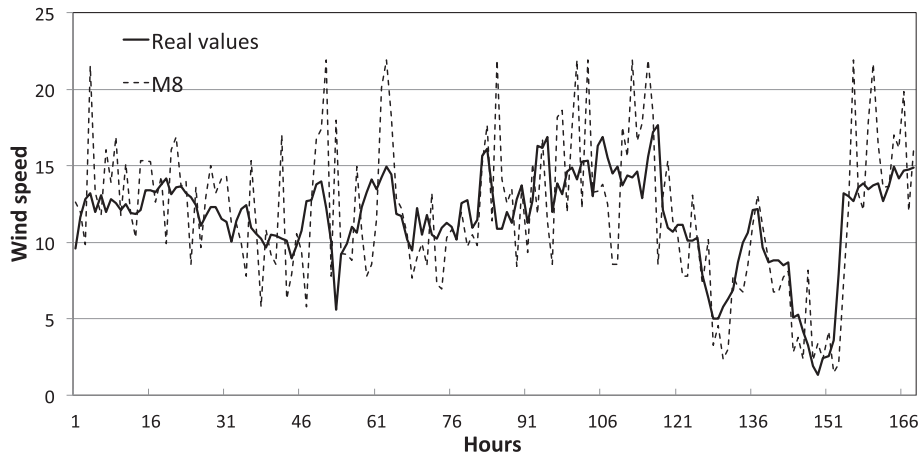


Fig. 10. Worst prediction in a week of the test set.

Table 7
Learning and forecasting time for regression trees with different models in leaf nodes.

	CPU time (in seconds)	
	Learning (training) time	Forecasting time
M1	0.017	0.000
M2	0.016	0.000
M3	0.022	0.000
M4	0.034	0.001
M5	0.019	0.000
M6	0.055	0.001
M7	0.056	0.001
M8	0.076	0.004

Table 8
MSE (in m/s) obtained for each tower by M6 and M7 regression trees and alternative algorithms for regression.

	ELM	GMDH	MLP	SVM	MLR	CART	CHAID	M6	M7
T1	1.391	1.395	1.370	1.378	1.408	1.463	1.613	1.338	1.353
T2	1.192	1.184	1.196	1.168	1.235	1.424	1.511	1.200	1.200
T3	1.286	1.273	1.243	1.240	1.305	1.324	1.561	1.235	1.235
T4	1.296	1.303	1.279	1.272	1.342	1.443	1.572	1.282	1.283
T5	1.492	1.455	1.432	1.410	1.516	1.514	1.595	1.385	1.385
T6	1.306	1.305	1.315	1.279	1.347	1.309	1.417	1.245	1.230
T7	1.771	1.785	1.745	1.768	1.834	1.722	1.747	1.608	1.605
T8	1.362	1.387	1.361	1.348	1.394	1.520	1.565	1.317	1.316
Avg	1.387	1.386	1.368	1.358	1.423	1.465	1.573	1.326	1.326

The best model for each tower is in bold.

Table 9
RMSE obtained for each tower by M6 and M7 regression trees and alternative algorithms for regression.

	ELM	GMDH	MLP	SVM	MLR	CART	CHAID	M6	M7
T1	0.191	0.191	0.184	0.191	0.188	0.210	0.256	0.169	0.173
T2	0.154	0.155	0.150	0.151	0.151	0.224	0.252	0.150	0.150
T3	0.205	0.202	0.188	0.199	0.206	0.219	0.303	0.181	0.181
T4	0.193	0.198	0.182	0.187	0.187	0.241	0.285	0.182	0.182
T5	0.177	0.188	0.174	0.178	0.179	0.203	0.225	0.161	0.161
T6	0.287	0.200	0.197	0.193	0.193	0.201	0.235	0.175	0.171
T7	0.266	0.236	0.224	0.231	0.238	0.219	0.226	0.186	0.185
T8	0.162	0.171	0.158	0.161	0.165	0.205	0.218	0.145	0.145
Avg	0.204	0.193	0.182	0.186	0.188	0.215	0.250	0.169	0.169

The best model for each tower is in bold.

Table 10
MAE (in m/s) obtained for each tower by M6 and M7 regression trees and alternative algorithms for regression.

	ELM	GMDH	MLP	SVM	MLR	CART	CHAID	M6	M7
T1	1.033	1.068	1.023	1.028	1.029	1.131	1.251	1.017	1.019
T2	0.880	0.880	0.872	0.871	0.876	1.099	1.182	0.898	0.898
T3	0.962	0.956	0.920	0.961	0.970	1.016	1.204	0.927	0.927
T4	0.966	0.974	0.939	0.957	0.946	1.105	1.204	0.964	0.964
T5	1.082	1.118	1.061	1.082	1.084	1.162	1.247	1.054	1.054
T6	1.265	0.987	0.973	0.966	0.971	0.998	1.087	0.931	0.922
T7	1.496	1.351	1.288	1.304	1.332	1.318	1.337	1.204	1.204
T8	1.007	1.034	0.988	1.001	1.021	1.178	1.230	0.982	0.982
Avg	1.086	1.046	1.008	1.021	1.029	1.126	1.218	0.997	0.996

The best model for each tower is in bold.

Table 11

RMAE obtained for each tower by M6 and M7 regression trees and alternative algorithms for regression.

	ELM	GMDH	MLP	SVM	MLR	CART	CHAID	M6	M7
T1	0.394	0.408	0.390	0.391	0.392	0.470	0.478	0.383	0.384
T2	0.358	0.359	0.355	0.355	0.357	0.490	0.481	0.354	0.354
T3	0.420	0.418	0.401	0.419	0.423	0.489	0.522	0.390	0.390
T4	0.408	0.412	0.396	0.404	0.400	0.527	0.507	0.396	0.396
T5	0.391	0.405	0.384	0.392	0.393	0.451	0.451	0.371	0.371
T6	0.518	0.416	0.410	0.407	0.409	0.455	0.456	0.381	0.377
T7	0.481	0.445	0.424	0.429	0.438	0.474	0.440	0.391	0.391
T8	0.360	0.370	0.354	0.358	0.365	0.467	0.440	0.342	0.342
Avg	0.416	0.404	0.389	0.394	0.397	0.478	0.472	0.376	0.376

The best model for each tower is in bold.

Table 12

R^2 obtained for each tower by M6 and M7 regression trees and alternative algorithms for regression.

	ELM	GMDH	MLP	SVM	MLR	CART	CHAID	M6	M7
T1	0.811	0.809	0.816	0.813	0.814	0.792	0.753	0.824	0.823
T2	0.847	0.845	0.851	0.850	0.849	0.779	0.753	0.837	0.841
T3	0.796	0.799	0.813	0.801	0.795	0.783	0.704	0.810	0.810
T4	0.807	0.802	0.818	0.813	0.814	0.759	0.721	0.810	0.810
T5	0.824	0.816	0.827	0.823	0.822	0.798	0.779	0.826	0.831
T6	0.806	0.803	0.805	0.807	0.809	0.799	0.775	0.817	0.823
T7	0.773	0.772	0.777	0.770	0.764	0.781	0.775	0.809	0.809
T8	0.839	0.833	0.843	0.841	0.836	0.796	0.787	0.846	0.840
Avg	0.813	0.810	0.819	0.815	0.786	0.837	0.756	0.822	0.823

The best model for each tower is in bold.

Table 13

IA obtained for each tower by M6 and M7 regression trees and alternative algorithms for regression.

	ELM	GMDH	MLP	SVM	MLR	CART	CHAID	M6	M7
T1	0.948	0.944	0.948	0.948	0.948	0.941	0.929	0.951	0.951
T2	0.958	0.957	0.959	0.959	0.960	0.937	0.929	0.955	0.956
T3	0.942	0.943	0.946	0.942	0.941	0.937	0.910	0.947	0.947
T4	0.946	0.944	0.948	0.946	0.949	0.929	0.917	0.946	0.946
T5	0.951	0.949	0.951	0.951	0.950	0.942	0.937	0.951	0.953
T6	0.899	0.939	0.941	0.945	0.942	0.943	0.936	0.949	0.951
T7	0.924	0.936	0.933	0.932	0.927	0.936	0.935	0.945	0.945
T8	0.956	0.954	0.957	0.956	0.954	0.941	0.940	0.958	0.956
Avg	0.940	0.946	0.948	0.947	0.946	0.938	0.929	0.950	0.951

The best model for each tower is in bold.

Multi-Linear Regression (MLR) [47], CART [21] and CHAID [48] are the algorithms considered for this comparison. The same training and test sets described in the previous section have been considered to obtain the model and the forecasts for each one of these methods. Tables 8–13 show this comparison of performance. It is possible to see that the proposed regression trees improve the

Table 14

Learning time by M6 and M7 regression trees and alternative algorithms for regression.

	CPU time (in seconds)
ELM	0.032
GMDH	0.858
MLP	17.773
SVM	511.437
MLR	0.025
CART	0.016
CHAID	1.125
M6	0.055
M7	0.056

forementioned techniques in this problem of very short-term wind speed prediction for all towers except for towers 2, 3 and 4 in which the SVM or MLP depending on the error obtain the best results.

However, SVM and MLP present the largest training times, and therefore, they could not be used for very short term problems. It can be concluded that the hybrid models that combine regression trees and nearest neighbors techniques provide promising results for the very short-term forecasting of wind speed in wind farms.

Finally, Table 14 shows the computing time for M6 and M7 regression trees and ELM, GMDH, MLP, SVM, MLR, CART and CHAID in order to present the lowest and highest training time.

4. Conclusions

In this paper we have shown that regression trees based on local models are a powerful methodology to take into account in wind speed prediction problems using wind data from measuring towers. The analysis has been based on the experimental comparison of eight types of RTs algorithms, including linear and non-linear approaches, in a real problem of very short-term wind speed prediction in a Spanish wind farm. We have also shown that RTs present a small computational burden, which makes easy the retraining of the algorithms if new wind speed data are collected in the measuring towers. We have extended the discussion by comparing the best RTs approaches in this problem against state-of-the-art regression algorithms, such as different types of neural networks (multi-layer perceptrons, extreme learning machines and GMDH networks), support vector regression algorithms, a multi-linear regression approach and other classical RTs, obtaining excellent results with the RT methodology.

Acknowledgments

The authors would like to thank Spanish Ministry of Science and Technology, Junta de Andalucía and Pablo de Olavide University for the support under projects ECO2010-22065-C03-02 and TIN2011-28956-C02, P12-TIC-1728 and APPB813097, respectively.

References

- [1] Lackner MA, Rogers AL, Manwell JF. The round robin site assessment method: a new approach to wind energy site assessment. *Renew Energy* 2008;33(9):2019–26.
- [2] Zhang W, Wang J, Wang J, Zhao Z, Tian M. Short-term wind speed forecasting based on a hybrid model. *Appl Soft Comput* 2013;13(7):3225–33.
- [3] Saavedra-Moreno B, Salcedo-Sanz S, Carro-Calvo L, Gascón-Moreno J, Jiménez-Fernández S, Prieto L. Very fast training neural-computation techniques for real measure-correlate-predict wind operations in wind farms. *J Wind Eng Ind Aerodynamics* 2013;116:49–60.
- [4] Lackner MA, Rogers AL, Manwell JF, McGowan Jon G. A new method for improved hub height mean wind speed estimates using short-term hub height data. *Renew Energy* 2010;35(10):2340–7.
- [5] Kalogirou SA. Artificial neural networks in renewable energy systems applications: a review. *Renew Sustain Energy Rev* 2001;5(4):373–401.
- [6] Öztopal A. Artificial neural network approach to spatial estimation of wind velocity data. *Energy Convers Manag* 2006;47(4):395–406.
- [7] Bilgili M, Sahin B, Yasar A. Application of neural networks for the wind speed prediction of target station using reference stations data. *Renew Energy* 2007;32(14):2350–60.
- [8] Velázquez S, Carta JA, Matías JM. Influence of the input layer signals of ANNs on wind power estimation for a target site: a case study. *Renew Sustain Energy Rev* 2011;15(3):1556–66.
- [9] Velázquez S, Carta JA, Matías JM. Comparison between ANNs and linear MCP algorithms in the long-term estimation of the cost per kWh produced by a wind turbine at a candidate site: a case study in the Canary Islands. *Appl Energy* 2011;88(11):3869–81.
- [10] Kramer O, Gieseke F, Satzger B. Wind energy prediction and monitoring with neural computation. *Neurocomputing* 2013;109:84–93.
- [11] Sheela KG, Deepa SN. Neural network based hybrid computing model for wind speed prediction. *Neurocomputing* 2013;122:425–9.

- [12] Barbounis TG, Theocharis JB. A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation. *Neurocomputing* 2007;70(7–9):1525–42.
- [13] Grassi G, Vecchio P. Wind energy prediction using a two-hidden layer neural network. *Commun Nonlinear Sci Numer Simul* 2010;15(9):2262–6.
- [14] Mohandes MA, Halawani TO, Rehman S, Hussain AA. Support vector machines for wind speed prediction. *Renew Energy* 2004;29(6):939–47.
- [15] Abdel-Aal RE, Elhadidy MA, Shaahid SM. Modeling and forecasting the mean hourly wind speed time series using GMDH-based abductive networks. *Renew Energy* 2009;34(7):1686–99.
- [16] Carta JA, Velázquez S, Matías JM. Use of Bayesian networks classifiers for long-term mean wind turbine energy output estimation at a potential wind energy conversion site. *Energy Convers Manag* 2011;52(2):1137–49.
- [17] Carta JA, Velázquez S. A new probabilistic method to estimate the long-term wind speed characteristics at a potential wind energy conversion site. *Energy* 2011;36(5):2671–85.
- [18] Beccali M, Cirrincione G, Marvuglia A, Serporta C. Estimation of wind velocity over a complex terrain using the generalized mapping regressor. *Appl Energy* 2010;87(3):884–93.
- [19] Fakhari A, Moghadam AM. “Combination of classification and regression in decision tree for multi-labeling image annotation and retrieval. *Appl Soft Comput* 2013;13(2):1292–302.
- [20] Kenesei T, Abonyi J. Hinging hyperplane based regression tree identified by fuzzy clustering and its application. *Appl Soft Comput* 2013;13(2):782–92.
- [21] Breiman L, Friedman JH, Olshen RA, Stone CJ. *Classification and regression trees*. Belmont CA: Wadsworth; 1984.
- [22] Quinlan JR. Learning with continuous classes. In: *Proceedings of the 5th Australian joint conference on artificial intelligence*; 1992. p. 343–8.
- [23] Torgo L. Functional models for regression trees leaves. In: *Proceedings of the 14th international conference on machine learning*; 1997. p. 385–93.
- [24] Kordos M, Piotrowski J, Bialka S, Blachnik M, Golak S, Wiczorek T. Evolutionary optimized forest of regression trees: application in metallurgy. In: *Proceedings of the 7th international conference on hybrid artificial intelligent systems*; 2012. p. 409–20.
- [25] Sela RJ, Simonoff JS. RE-EM trees: a data mining approach for longitudinal and clustered data. *Mach Learn* 2012;86(2):169–207.
- [26] Fidalgo-Merino R, Nuñez M. Self-adaptative induction of regression trees. *IEEE Trans Pattern Analysis Mach Intell* 2011;33(8):1659–72.
- [27] Young AW, Weckman RG, Hari V, Whiting H, Snow A. Using artificial neural networks to enhance CART. *Neural Comput Appl* 2012;21(7):1477–89.
- [28] Toth D, Eltinge JL. Building consistent regression trees from complex sample data. *J Am Stat Assoc* 2011;106(496):1626–36.
- [29] Chang Y. Variable selection via regression trees in the presence of irrelevant variables. *Commun Statistical Simulation Comput* 2013;42(8):1703–26.
- [30] Haslinger R, Pipa G, Lewis LD, Nikolic D, Williams Z, Brown E. Encoding through patterns: regression tree-based neural population models. *Neural Comput* 2013;25(8):1953–93.
- [31] Aho T, Zenko B, Dzeroski S, Elomaa T. Multi-target regression with rule ensembles. *J Mach Learn Res* 2012;13:2367–407.
- [32] Tsiros IX, Dimopoulos IF, Chronopoulos KI, Chronopoulos G. Estimating airborne pollutant concentrations in vegetated urban sites using statistical models with microclimate and urban geometry parameters as predictor variables: a case study in the city of Athens Greece. *J Environ Sci Health* 2009;44(4):1496–502.
- [33] Smith PF, Ganesh S, Liu P. A comparison of random forest regression and multiple linear regression for prediction in neuroscience. *J Neurosci Methods* 2013;220:85–91.
- [34] Aertsen W, Vincent K, De Vos B, Deckers J, Orshoven JV, Muys B. “Predicting forest site productivity in temperature lowland from forest floor, soil and litter fall characteristics using boosted regression trees. *Plant Soil* 2012;354: 157–72.
- [35] Leclere J, Oberdorff T, Belliard J, Leprieux F. A comparison of modeling techniques to predict juvenile 0+ fish species occurrences in a large river system. *Ecol Informatics* 2011;6(5):276–85.
- [36] Choi W, Paulson SE, Casmassi J, Winer AM. Evaluating meteorological comparability in air quality studies: classification and regression trees for primary pollutants in California's South Coast Air Basin. *Atmos Environ* 2013;64:150–9.
- [37] Garner GG, Thompson AM. Ensemble statistical post-processing of the National Air Quality Forecast capability: enhancing ozone forecasts in Baltimore, Maryland. *Atmos Environ* 2013;81:517–22.
- [38] Mori H, Awata A. Feature extraction of meteorological data using regression tree for wind power generation. In: *Proceedings of the IEEE international conference on sustainable energy technologies*; 2008. p. 1104–7.
- [39] Clifton A, Kilcher L, Lundquist JK, Fleming P. Using machine learning to predict wind turbine power output. *Environ Res Lett* 2013;8.
- [40] Fugon L, Juban J, Kariniotakis J. Data mining for wind power forecasting. In: *Proceedings of the European wind energy conference and exhibition*; 2008. p. 2817–22.
- [41] Package CORElearn available in software R: <http://lkm.fri.uni-lj.si/rmarko/software>.
- [42] Liu Hui, Chen Chao, Tian Hong-qi, Li Yan-fei. A hybrid model for wind speed prediction using empirical mode decomposition and artificial neural networks. *Renew Energy* 2012;48:545–56.
- [43] Ramirez-Rosado Ignacio J, Alfredo Fernandez-Jimenez L, Monteiro Cláudio, Sousa Joao, Bessa Ricardo. Comparison of two new short-term wind-power forecasting systems. *Renew Energy* 2009;34:1848–54.
- [44] Huang GB, Zhu QY, Siew CK. Extreme learning machine: theory and applications. *Neurocomputing* 2006;70(1–3):489–501.
- [45] Bishop CM. *Neural networks for pattern recognition*. Oxford University Press; 1995.
- [46] Smola AJ, Schölkopf B. A tutorial on support vector regression. *Statistics Comput* 2004;14(3):199–222.
- [47] Brown SH. *Multiple linear regression analysis: a matrix approach with MATLAB*. Ala J Math Spring/Fall, 2009:1–3.
- [48] Kass Gordon V. An exploratory technique for investigating large quantities of categorical data. *Appl Stat* 1980;29(2):119–27.