Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

Estimation of wind velocity over a complex terrain using the Generalized Mapping Regressor

M. Beccali [a], G. Cirrincione [b], A. Marvuglia [a,*], C. Serporta [c]

[a] Dipartimento di Ricerche Energetiche ed Ambientali (DREAM), Università degli Studi di Palermo, Viale delle Scienze – edificio 9, 90128 Palermo, Italy
[b] Department de Génie Électrique, Universitè de Picardie Jules Verne, 33, Rue Saint Leu, 80039 Amiens, France
[c] ISSIA-CNR (Institute on Intelligent Systems for the Automation), Section of Palermo, Via Dante12, Palermo, Italy

ARTICLE INFO

Article history:
Received 9 January 2008
Received in revised form 9 May 2009
Accepted 19 May 2009
Available online 13 June 2009

Keywords:
Neural networks
Generalized Mapping Regressor
Curvilinear Component Analysis
Wind
Sicily

ABSTRACT

Wind energy evaluation is an important goal in the conversion of energy systems to more environmentally friendly solutions. In this paper, we present a novel approach to wind speed spatial estimation on the isle of Sicily (Italy): an incremental self-organizing neural network (Generalized Mapping Regressor – GMR) is coupled with exploratory data analysis techniques in order to obtain a map of the spatial distribution of the average wind speed over the entire region.

First, the topographic surface of the island was modelled using two different neural techniques and by exploiting the information extracted from a digital elevation model of the region. Then, GMR was used for automatic modelling of the terrain roughness. Afterwards, a statistical analysis of the wind data allowed for the estimation of the parameters of the Weibull wind probability distribution function. In the last sections of the paper, the expected values of the Weibull distributions were regionalized using the GMR neural network.

1. Introduction and state of the art

The increasing demand for energy, coupled with the shortage of traditional energy sources, has accelerated the research efforts of the scientific and industrial world towards the efficient exploitation of renewable sources of energy. Knowledge of the spatial distribution of wind speed is essential for assessing the energy output of a regional wind energy conversion system. The use of wind energy in Sicily today shows promising potential due to the large availability of the resource and the many political and economic opportunities. The potential benefits of a reliable wind speed prediction and mapping model are obvious in wind power generation; however, there are many cases in which wind data do not exist for a particular area, but a quick assessment of wind potential is needed.

At present, when wind data are not available for a particular site, the selection of the wind farm site relies on data from the nearest measuring station and on a wind flow analysis that takes into account the topography and the roughness level of the surrounding land. However, these calculations may result in significant errors in estimating the wind speed, which lead to even higher errors in energy estimation, especially over complex terrains [1]. A short but effective review of the existing approaches for wind resource estimation has been performed by Landberg et al. [2]. One of the approaches often used in estimating the wind resource at a site is the measure-correlate-predict (MCP) method [3], which uses a short measuring campaign at the site and then correlates these measurements with an overlapping time series of another site using simple statistical models. However, these models usually overestimate or underestimate the wind potential.

Several physical models based on the use of weather data have also been developed for wind speed forecasting and wind power predictions [4]. These advanced numeric weather prediction (NWP) models have the potential to improve the modelling of wind flow, particularly in complex terrain. However, since they are very complex, they are usually run on supercomputers, which limit their usefulness for the on-line or very-short-term operation of power systems. Other methods use global databases of meteorological measurements or atmospheric mesoscale models, but they require large computational systems in order to achieve accurate results [2].

The numerical codes for wind field modelling over rough terrain are generally divided into two types [5]: *dynamic models* (also called *prognostic*) and *kinematic models* (also called *diagnostic*). The first ones are based on the solution of a full set of time-dependent environmental fluid mechanics equations. The second generate a wind field by satisfying some physical constraints [6,7]. In all of these cases, the first step is to estimate the mean value of the wind speed that is expected at a site and afterwards to estimate the wind energy that a proposed wind farm would produce in an average year.

[*] Corresponding author. Tel.: +39 091236139; fax: +39 091484425.
E-mail address: marvuglia@dream.unipa.it (A. Marvuglia).

0306-2619/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.apenergy.2009.05.026

The task addressed by this paper is the spatial interpolation and mapping of wind data that is representative of the yearly wind regime of the investigated region. The topic of temporal forecasting of the wind time series was not dealt with in this work. Forecasting of wind turbine power outputs in Sicily, despite being an important and interesting topic, was also not approached in this paper because of the lack of data about wind farm power outputs.

In this paper, we describe the application of a novel integrated approach in which different techniques are used for spatial wind speed estimation above a complex terrain surface such as that of Sicily. The task of this work is twofold: to supply the regional energy planners with a useful tool for choosing the site of wind farms and to present a general methodology that can be applied in any other regional context to achieve the same goal. The approach applied in this paper is different than the ones described above: it is a typical data-driven approach, essentially based on the use of artificial neural networks (ANNs). This means that no physical or mathematical law connecting the variables at hand is used, because the model lets the data "speak for themselves", or, in other words, the learning process is based on the data themselves. ANNs have been widely used in different fields including pattern recognition, approximation, and time series prediction. Nowadays they are also widely used for spatial data interpolation in an attempt to overcome some of the limitations of more traditional spatial analysis methods. A sufficiently wide, though not exhaustive, overview of some of the neural, geostatistical and hybrid models used for space-temporal wind forecasting was already provided in the paper by Cellura et al. [8]. The most commonly used neural model is the multi-layer perceptron (MLP) network, but other models such as radial basis functions (RBF), counter-propagation neural networks (CPNN) and hybrid models have also been used. ANNs offer the advantage that they contain no critical assumptions about the nature of the spatial data to be processed and are well suited to process noisy and non-linear data manifolds, which is the type of data generated in environmental studies.

2. Theoretical background

ANNs consist of numerous simple *processing units* (called *neurons*) that we can globally program to *learn* the underlying relations contained in the data. For this reason, in order to correctly model the studied system, neurons need to be *trained* with a *training set* (TS), which must be representative of the intrinsic dynamics of the system. In the following sections, a brief description of the main neural techniques used in this work will be provided, and the reader is forwarded to the cited references for deeper investigation.

2.1. Curvilinear Component Analysis

Curvilinear Component Analysis (CCA) [9] is a very powerful data analysis method, conceived to extract relevant information from data. This method makes it possible to determine the so-called *intrinsic dimension* of a data set, i.e., the smallest number of variables that are needed to describe the set of data without any significant information loss. CCA is a self-organizing neural network able to provide revealing low-dimensional mapping of a high-dimensional and non-linearly-related data set. In summary, the algorithm proceeds to a *global unfolding* followed by a *local projection* onto the average manifold of the data (Fig. 1). Let us consider an input consisting of N_s samples belonging to some p-dimensional manifold, embedded in an n-dimensional input space $\mathbf{X} = \{x_{ik}\}$ $i = 1 \ldots N_s$, $k = 1, \ldots, n$.

If the data set comes from a non-ideal process, it is generally noisy and the manifold has some "thickness", being thus also of dimension n. The aim of the CCA is to find the underlying manifold of data (the *average manifold*) and to map it onto a p-dimensional output space Y. In order to do this, the algorithm uses N_n neurons with n-dimensional input weights and p-dimensional output weights. The Euclidean distances $X_{ij} = d(x_i, x_j)$ between all of the input vectors x_i are considered. CCA forces the distances $Y_{ij} = d(y_i, y_j)$ between the corresponding output vectors y_i to match X_{ij} for each possible pair (i, j). This is accomplished by minimizing a cost function [11].

In order to check the preservation of the topology of the CCA projection, it is possible to use a representation that is called " dY–dX". It consists of the joint distribution of input and output distances between pairs of neurons. For each possible pair of neurons, a point is plotted on the dY–dX plane, where dX is the distance between the neurons on the input layer, and dY is the distance between the corresponding neurons on the output layer. In a topology preserving mapping, dY should be proportional to dX, at least for small dY distances. In this representation, a locally correct mapping is shown by a straight line with a slope equal to one near the origin of the axes. On the other hand, strong unfolding is revealed by curvature and spreading of the points plotted on the dY–dX plane.

2.2. The Generalized Mapping Regressor: a short description

The Generalized Mapping Regressor (GMR) is an incremental self-organizing neural network with chains (second layer weights) among neurons. It transforms the mapping problem into a pattern recognition problem by working in the *augmented space*, where vectors are created by attaching the corresponding output vector to the input vector. In the *augmented space*, the branches of the function or the relation that one wants to approximate become clusters that have to be identified by GMR.

The algorithm is here described in a qualitative way (it is detailed in [12]). It consists of four phases: training, linking, merging, and recall.

The algorithm EXIN SNN [13] is used for the training; it recovers the augmented space by either creating neurons or adapting their weights according to the novelty of the input data. This novelty is quantified by a threshold, called *vigilance threshold* (ρ), which determines the resolution of the modelling. The neurons created after the first epochs (i.e., presentations of the whole TS to the network) are called *object neurons* (*coarse quantization*). Subsequently, the data are labelled according to the nearest object neuron. In the following epochs (*fine quantization*), as many EXIN SNNs as objects are trained by using data with the same label as the TS. A pool of neurons (*final neurons*) is found. Once trained, all data are relabelled according to the nearest final neuron (based on a smaller vigilance threshold than in the case of coarse quantization). The set of all data labelled according to the same neuron are defined as the *neuron domain*.

The next phase is the linking phase. Linking neurons is accomplished by computing the second layer weights, which are discrete and equal to zero in the absence of a link. A link is computed at the presentation of each datum of the TS. Originally, this computation was based on the requirement that connected neurons must approach the direction (called *linking direction*, LD) of the vector connecting the input vector to the weight of the winning[1] neuron in the *augmented space* [13,14]. At each presentation of an input vector, the neurons that are candidates to be linked to the winning neuron are sought inside a hypersphere centered on the winning neuron whose radius is equal to ∂ times the distance between the input vector and the winning neuron. Alternatively, it is possible to choose in

[1] That is, the neuron whose weight vector is the closest to the input vector.

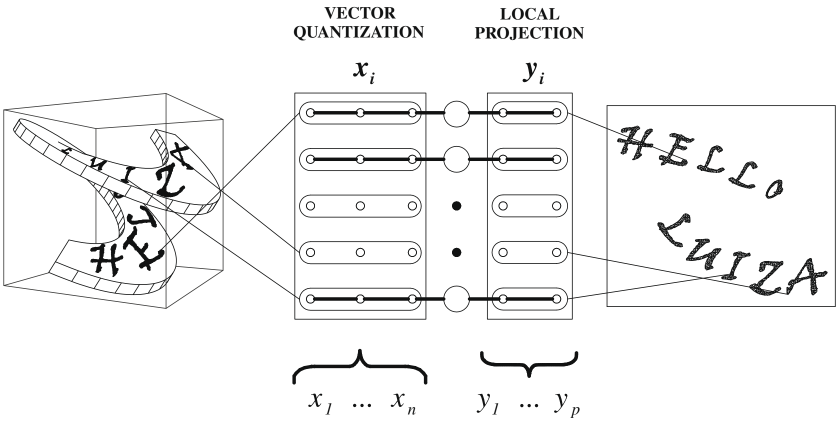

Fig. 1. Principle of the CCA algorithm. The input weights first proceed to a vector quantization [10] of the input data space (X) in n dimensions. Then, the output weights project the input average manifold onto an output representation space (Y) of dimension $p < n$. Here a two-dimensional mapping of a text which is folded in a three-dimensional space is done. In practice this algorithm reveals the underlying submanifold of a data set through "unfolding" and dimension reduction, leading to the concept of CCA.

advance the number of candidates k (k-Test). Hence, this linking carries directional information driven by the input data in order to determine both the position and the shape of the mapping branch (*cluster*). However, because the TS is always noisy and the input data can be placed everywhere around the neuron, the linking direction does not sufficiently represent the branch shape. A better technique, which exploits the domain *principal directions* (PDs), is also employed [12]. For each datum, the weights are sorted according to the Euclidean distance from the datum, and the winning neuron is determined. This one is then linked to another neuron chosen in a subset of the neuron pool (*candidate neurons*). Two criteria have been implemented for determining the subset. The first (δ-BnB) considers only the neurons included in a hypersphere centred on the data cloud whose radius is a multiple (defined a priori by a *linking factor*) of the distance between the input and the winner weight vectors. The second (k-BnB) considers only the k nearest neighbours of the input. In the latter criterion, the value of k has to be defined in advance. Then, for each candidate, the absolute value of the scalar product between its PD and the winner's PD is evaluated. The winner is linked to the candidate yielding the maximum scalar product (i.e., the candidate whose PD is closest in direction to the winner's PD). However, this kind of linking is not flexible with respect to noise. In the paper by Cirrincione et al. [12], the authors present a different linking algorithm that is more suitable for noisy databases.

In the merging phase, GMR checks whether different objects are linked. If they are, the objects are merged. The recall phase replaces the neurons in the reduced manifold with Gaussians representing the domain. Their parameters are estimated by the maximum likelihood (ML) technique. Simple tests and a Gaussian kernel interpolation determine all of the possible outputs of the network.

3. Case study: wind speed spatial estimation

The data used for the study are represented by the hourly mean values of wind speed at 10 m above the ground level (a.g.l.), recorded at 29 different anemometric stations spread out on the Sicilian territory.

Fig. 2 shows the locations of the 29 anemometric stations marked with small squares. The data from the Fiumedinisi station were used only for validation of the obtained results.

3.1. Topography model

The base information for the modelling operation described in this section is represented by a digital elevation model (DEM) of the region with a sampling interval[2] of 30 m. In order to obtain the TS for the neural model of the terrain surface, a regular mesh of square elements with a cell size of 850 m was generated. This mesh was superimposed onto the DEM and the geographic coordinates (*Easting* and *Northing*) of its nodes were automatically extracted.

The approximation of the land surface of Sicily was obtained by following a hierarchical approach; the first level is represented by a pre-processing phase, and the second level consists of the actual modelling.

During the pre-processing phase, the data (namely, the three-dimensional vectors whose components are the *Easting*, *Northing*, and *Elevation* of each point of the mesh) were first linearly normalized within the interval [0, 1] and then clustered by following a neighbourhood criterion, implemented through the utilization of a self-organizing map (SOM). SOMs are a particular kind of NN used for data clustering purposes [15]. Each neuron of a SOM is represented by a weight vector (*prototype vector* or *codebook vector*) whose dimension is the same as the dimension of the input vectors.

Several SOMs were tested and the obtained results were evaluated on the basis of the Davies–Bouldin (D–B) cluster validity index [16]. The chosen map was a *sheet shaped* SOM with a rectangular grid, and it was made up of 100 neurons. The *k-means* partitive clustering algorithm [17] was implemented on the *prototype vectors* of the trained SOM. The algorithm was applied repeatedly, with the number of clusters ranging from 2 to 20 (a reasonable number in the framework of our application), and the corresponding trend of the D–B index was observed. As the partition realized by the *k-means* algorithm is not unique, the algorithm was run 20 times for each fixed value of the number of clusters, and for that value the chosen partition was that with the minimum *quantization error,* defined as:

$$E_q = \sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{m}_{i^*}\| \tag{1}$$

where $\mathbf{m}_{i^*}$ is the weight vector associated with the winning neuron for the input vector $\mathbf{x}_i$, and $||\cdot||$ indicates the Euclidean distance.

The chosen cluster configuration identified six clusters. The obtained clusters are represented in Fig. 3, along with the neurons of the trained SOM. Fig. 4 shows the value of the D–B index as a func-

[2] The sampling interval of the DEM represents its level of detail (resolution), which is related to its cell size.

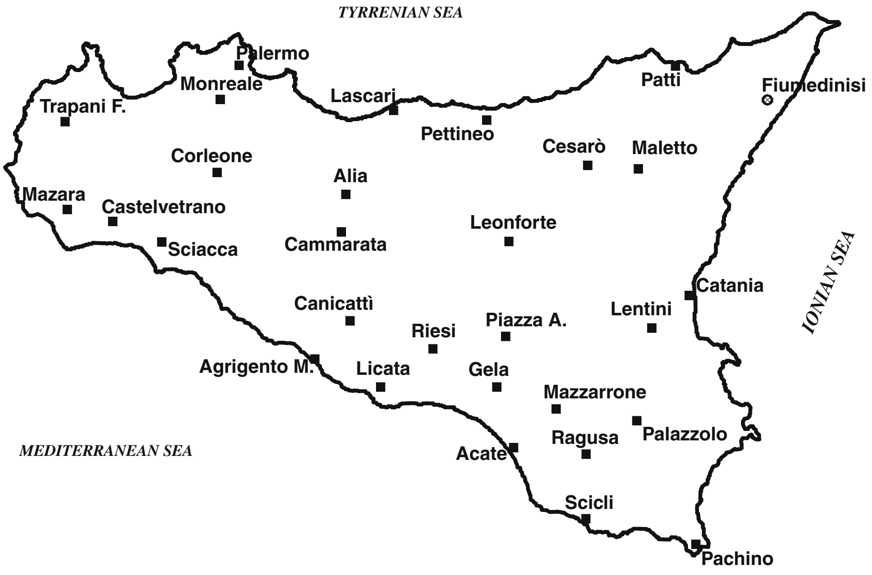

Fig. 2. Locations of the anemometric stations that supplied the wind data used for the study.

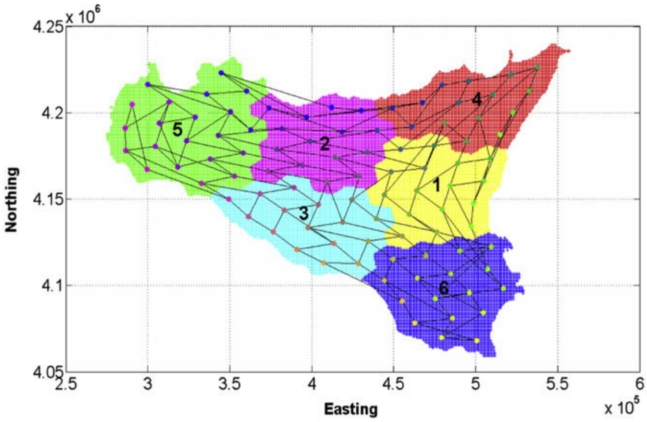

Fig. 3. Representation of the six clusters obtained by using a SOM with rectangular grid made up of 100 neurons (the coordinates of the plotted grid points are the de-normalized ones).

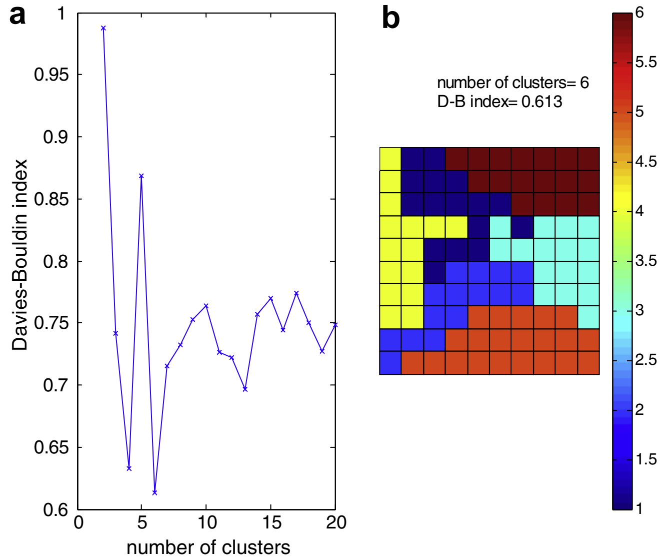

Fig. 4. Plotting of Davies–Bouldin (D–B) index vs number of clusters (a) and representation with different colours of the corresponding six clusters in which the map units of the SOM were subdivided (b).

tion of the number of clusters and a schematic representation of the clusters in which the *prototype vectors* of the SOM were subdivided through the *k-means* algorithm.

Using the points belonging to each of the six clusters, a different GMR was trained. In this case, the data were normalized in the interval [−1, +1] before training.

Fig. 5a and b shows the results of the linking phase obtained in the parts of the region corresponding to clusters 1 and 6, and to clusters 2, 3, and 5, respectively. The PD method with $k = 4$ was used as the linking method. In the rough phase, $\rho_1 = 0.5$ was used, while in the fine tuning phase, $\rho_2 = 0.05$ was used. As can be noted, the distribution of the neurons follows the data distribution very well.

3.2. Roughness model

One important factor that is able to influence the wind profile within the so-called *boundary layer* is the terrain roughness. In fact, it has a deep influence on the frictional forces acting on the air flow blowing over the ground surface. In particular, the influence of the surface roughness on the vertical wind profile is taken into account in some specific formulas through a term called the *surface roughness length*, which is strongly dependent on the land coverage within the investigated area. The *roughness length* of the whole region was modelled here by the implementation of another GMR model. In this case, only one GMR was used because the surface to be approximated is composed of discrete values, and it is not necessary to use a high number of neurons to model it and identify the discontinuities.

The information concerning the land coverage was obtained by the superimposition of the same mesh grid as before over the georeferred digital land cover map created by the CORINNE (coordination of the information on the environment) project of EC (updated in 2002). In order to assign a value of the *roughness length* to each part of Sicily, we used a table proposed by Wieringa [18]. In a nutshell, we reclassified the CORINNE land map of Sicily by applying a similarity criterion with the classes contained in Wieringa's table.

The PD method with $k = 4$ was used for the linking phase (with $\rho_1 = 0.5$ and $\rho_2 = 0.05$). Fig. 6a and b shows the results of the link-

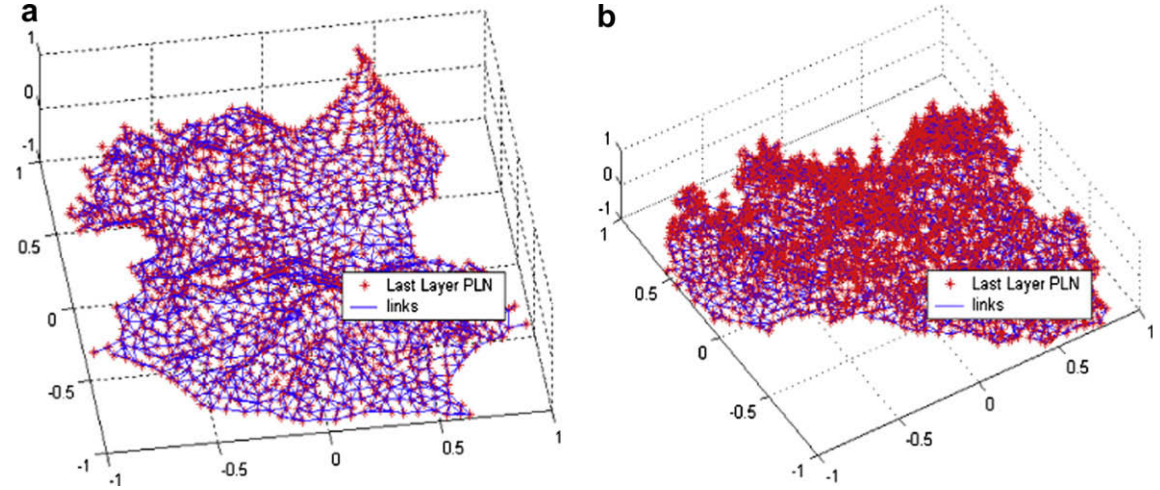

Fig. 5. Topography model. Results of the linking phase obtained by using the PD method with $k = 4$: part of Sicily corresponding to clusters 1 and 6 (a) and clusters 2, 3 and 5 (b).

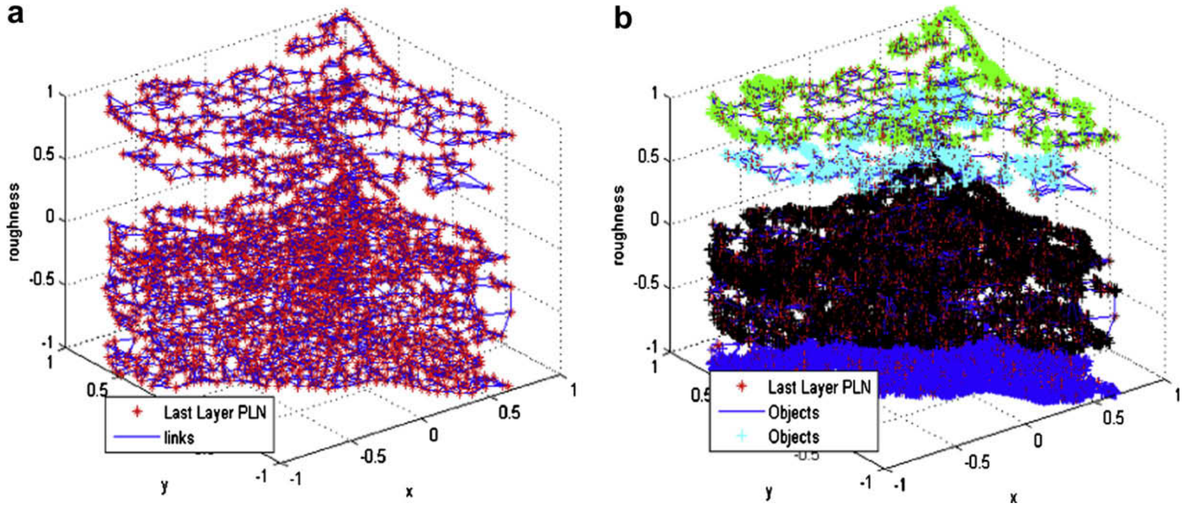

Fig. 6. Roughness model. Results of the linking phase (a) and of the merging phase (b).

ing and merging phases, respectively. The different "macro-typologies" of surface roughness appear in Fig. 6 as overlapping layers.

3.3. First approximation estimate of the wind speed

The base information for this phase of the analysis is represented by the expected values of the theoretical Weibull distribution functions describing the wind regime at each of the 29 measurement sites available, as determined in the paper by Cellura et al. [19]. However, because GMR is basically incremental, the number of neurons depends on the complexity of the problem and the learning parameters. If only a few (e.g., $N = 29$) observations are fed to GMR, the network would require only N neurons or less, which is insufficient for the spatial approximation. In order to solve this problem, a large number of neurons were induced by creating a fictitious TS; using the inverse distance weighting (IDW) technique, the mean wind speed was estimated at each point of the same mesh grid used in Sections 3.1 and 3.2. Despite the fact that the estimate obtained using this method is not very accurate, it is not supposed to have a negative effect on the reliability of the spatial modelling realized by GMR for reasons that will be explained in the following section. By using IDW, it was possible to estimate the values of the average wind speed at 10 m a.g.l. for each point of the grid, starting at the expected values of the Weibull distributions that had been previously computed.

3.4. Data investigation by CCA

In this section, the CCA technique is applied with the aim of identifying the *intrinsic dimension* of the data set under study and thus understanding how to pre-process (project) the data in order to create a TS for GMR. In fact, CCA is not used simply as a nonlinear projector, its diagrams are also used as tools for the detection and analysis of nonlinearities. The data set at hand, which we will call the *wind manifold* henceforth, is made up of five-dimensional vectors. We use five-dimensional vectors because, for each point of the above-mentioned mesh grid, one knows the three spatial coordinates, the value of the *roughness length,* and the average wind speed estimated using the IDW method.

The version of the CCA algorithm used here is characterized by a variable λ[3] which decreases proportionally to the inverse of the iteration step, from the initial value of $\lambda_{in} = 12$ up to $\lambda_{fin} = 0.05$.

[3] λ is a pre-defined distance the cost function depends upon (see [11]).

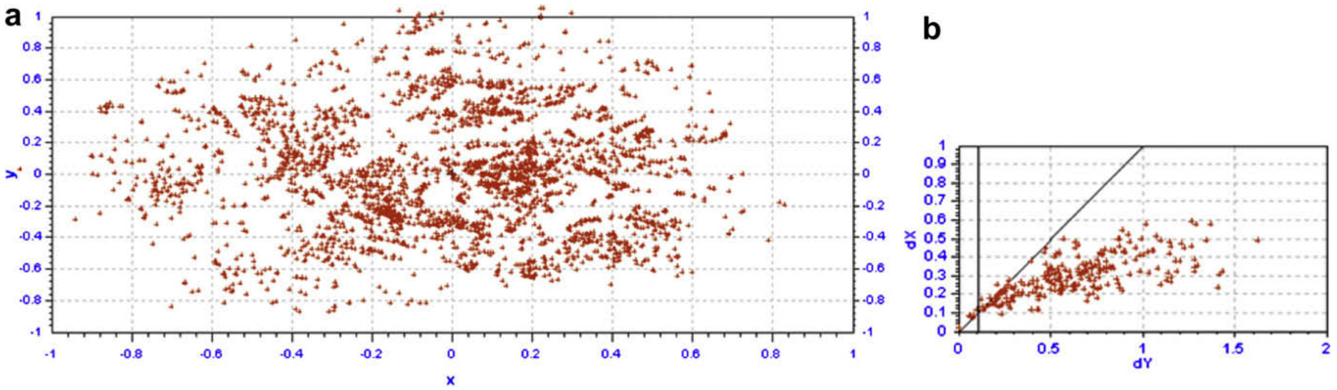

Fig. 7. Two-dimensional projection of the five-dimensional TS (a) and related *dY*–*dX* diagram (b).

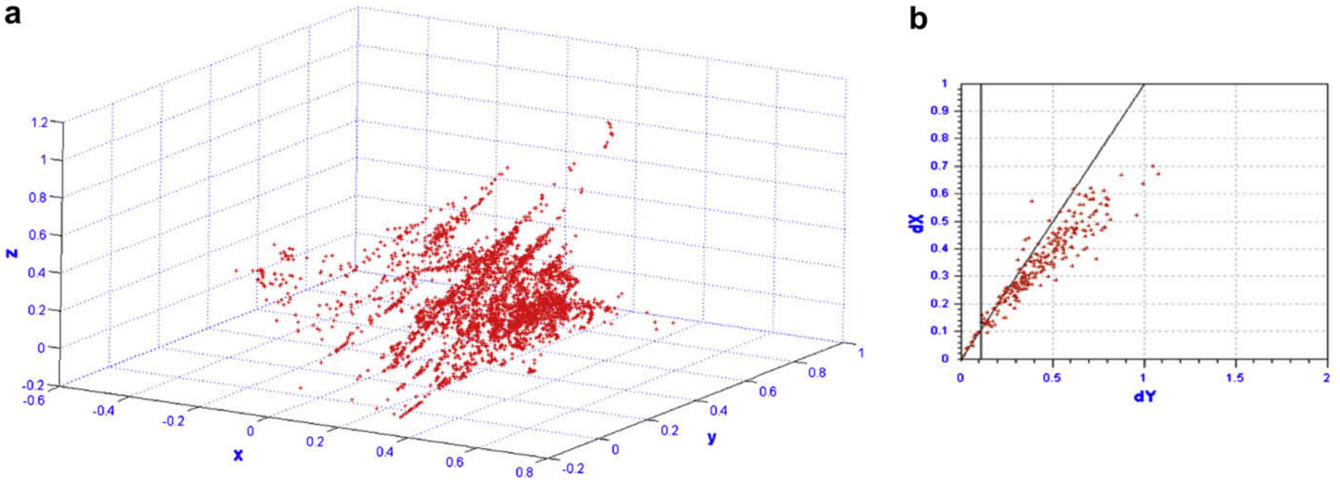

Fig. 8. Three-dimensional projection of the five-dimensional TS (a) and related *dY*–*dX* diagram (b).

Fig. 7a shows the projection of the five-dimensional data set onto a two-dimensional space, and Fig. 7b shows the corresponding *dY*–*dX* diagram. In this plot, it is possible to notice the presence of a strong nonlinearity (the bending of the data cloud with respect to the bisector), which indicates that dimension two does not fit the data. The projection onto a three-dimensional space (Fig. 8a and b shows a narrower scattering of the points in the *dY*–*dX* diagram, but it reveals that dimension three is still inadequate to correctly represent the data. Fig. 9a shows the points obtained by plotting the first two CCA components of the five-dimensional vectors projected onto a four-dimensional space. Fig. 9b shows the corresponding *dY*–*dX* diagram. The low scattering of the points close to the bisector suggests that the intrinsic dimension of the *wind manifold* is close to four. This fact implies that reducing data dimensionality by projecting the data in dimensions that are equal to the *intrinsic dimension* or lower, may result in a strong mutual link among the components, as will be better explained in Section 4.

4. Wind speed spatial estimation by GMR

In this section, the estimation of the mean hourly wind speed at 10 m a.g.l. will be described. The temporal information is resumed in the Weibull distributions related to each measurement site and only their expected values are retained (the corresponding variances are considered to be estimators of the error for the anemometric station). The expected values of the Weibull distributions were used instead of the sample means, because it is more reliable to work on the distribution than on the sample first order moments. The input space vectors have four components (the three spatial coordinates plus the *roughness length*), the output space is one-dimensional (the wind speed), and the augmented space is thus five-dimensional.

Two experiments were performed; in the first one, the *wind manifold* was pre-processed by a projection onto a four-dimensional space before the learning phase was run, and in the second one, the *wind manifold* was projected onto a three-dimensional space.

4.1. GMR with data projected in a four-dimensional space

Once the intrinsic dimension was estimated as described in Section 3.4, a TS for the GMR was obtained by projecting the original TS in a four-dimensional space using the principal component analysis (PCA, [20]) technique, which is faster than CCA. Afterwards, the data were statistically normalized. The EXIN SNN learning required $\rho_1 = 0.9$ and $\rho_2 = 0.3$. Pruning strategies were used in order to decrease the final number of neurons. Each learning phase lasted four epochs. Concerning the linking phase, both the LD and the PD methods with two candidates ($k = 2$) yielded too many objects after merging. Only two objects were recovered when using the PCA method with $k = 4$, which is the final choice for this project. The grid containing the territory of Sicily was divided into two sub-grids of the same size by picking out one point out of

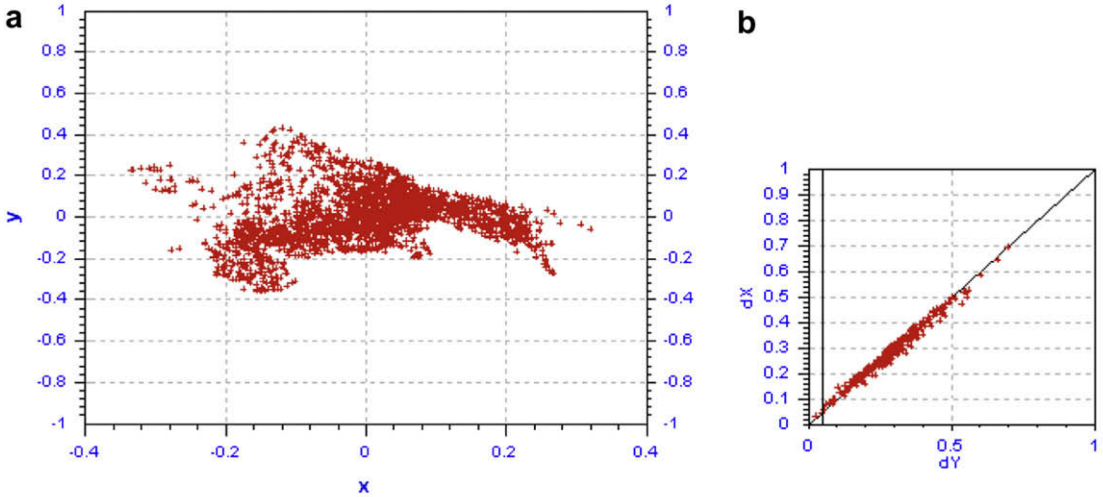

Fig. 9. Visualization in a two-dimensional space of the four-dimensional projection of the five-dimensional TS (a) and related *dY*–*dX* diagram (b).

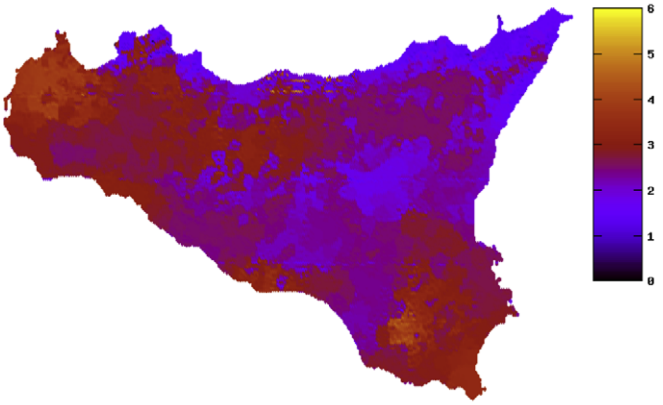

Fig. 10. Map of the average wind speed (m/s) at 10 m a.g.l. obtained with the validation set by the GMR network trained with data projected in a four-dimensional space.

two for the TS and leaving the other one for the data set to be used for results validation (*validation set*). The standard deviation of the Gaussian kernel used for the interpolation in the recall phase is approximately one-third of the range of the Gaussian-cosine experimental semi-variogram shown in the paper by Cellura et al. [19].

Fig. 10 shows the map obtained when the GMR is fed with the data of the validation set. The correlation coefficient between the residuals and the measured data is equal to 0.210, which means that most of the spatial data correlation was explained by the neural network.

The choice to use the data projected in a dimension that is close to the intrinsic dimension of the data manifold (but still slightly lower[4] than it) constrains all of the variables, forcing the wind speed to be linked to the other features. In other words, by projecting in a dimension that is slightly lower than the intrinsic dimension, the components of the original vectors are linked to each other in such a way that the variables known to have a higher level of precision (in this case, the spatial coordinates and, with some exception, the *roughness length*) drive the learning process of the features of those variables that are known to have a lower degree of precision (in this case, the average wind speed).

In a very large sense, the described approach can be seen as a non-linear *co-kriging* technique [21]. However, unlike *co-kriging*, it does not require the estimation of all the cross-covariance functions, which can be a computationally intensive task.

4.2. GMR with data projected in a three-dimensional space

Here, the original TS is projected in a three-dimensional space by PCA. The EXIN SNN learning requires $\rho_1 = 0.9$ and $\rho_2 = 0.2$, and pruning is allowed. Each learning phase lasts four epochs. Fig. 11a shows the PCA projections of the TS data (after normalization) along with the fine quantization neurons of GMR and their links. In the linking phase, the PCA method with four candidates ($k = 4$) yields, after the merging phase, only one object (see Fig. 11b). The recall phase is the same as in the previous subsection. The map obtained by feeding GMR with the data of the validation set is shown in Fig. 12. This map is similar to that shown in Fig. 10, but smoother (because dimension three is further from the intrinsic dimension), and the representation is less accurate.

The correlation coefficient between the residuals and the measured data in this case is equal to 0.169, which means that nearly all of the spatial data correlation was explained by the neural network. Indeed, the remaining correlation is due to noise in the data and modelling errors.

4.3. Estimation of the average wind speeds at 50 m above the ground level

The evaluation of the yearly theoretical power output of a wind turbine at a certain location depends upon the curve of the wind speed distribution at the turbine's hub height. For many commercial turbines, the hub's height is 50 m, although more powerful and higher turbines are being commercialized at present.

In this paper, the average wind speed at 50 m a.g.l. was obtained by the application of the following formula [22]:

$$\frac{U_z}{U_{z_{ref}}} = \left(\frac{z}{z_{ref}}\right)^{\alpha} \tag{2}$$

where U_z is the wind speed at height z, $U_{z_{ref}}$ is the reference wind speed at height z_{ref}, and α is the power law exponent. In this case, we used a height of 10 m a.g.l. as z_{ref}.

The exponent α is a highly variable quantity. Some researchers have developed methods for calculating α [22]. The expression used in this paper is the one proposed by Justus [23]:

[4] See Fig. 9: the points cloud in the *dY*-*dX* diagram still has a small "thickness" around the bisector.

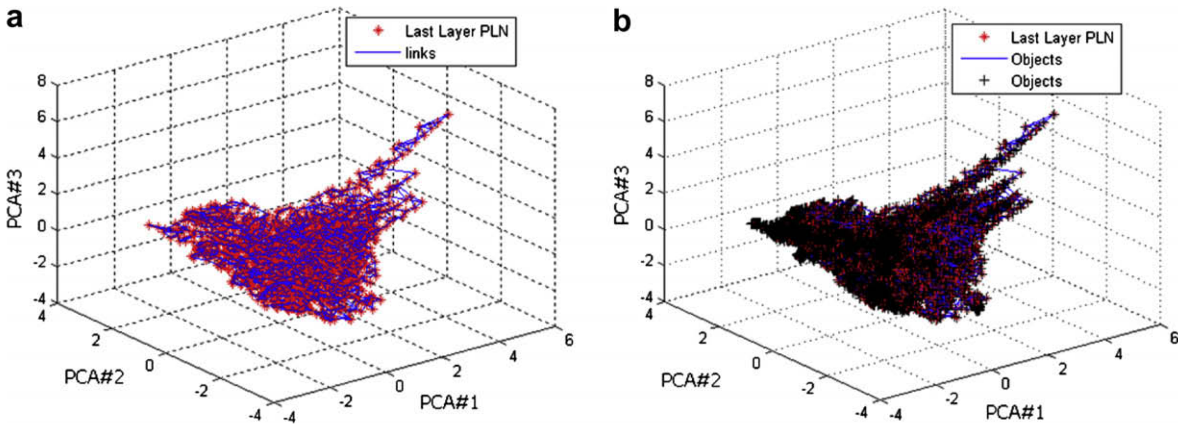

Fig. 11. PCA-projected TS, fine quantization neurons and links (a) result of the GMR merging phase (b).

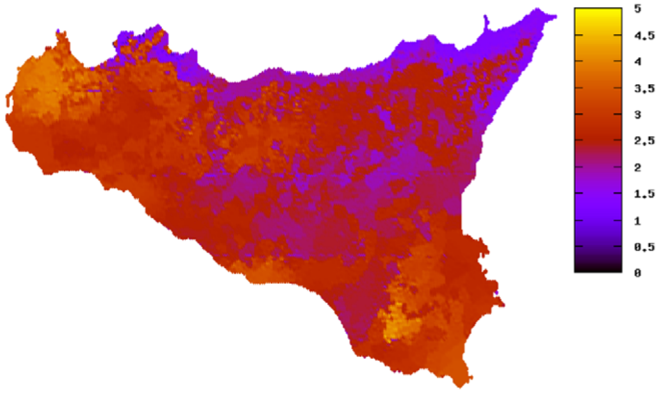

Fig. 12. Map of the average wind speed (m/s) at 10 m a.g.l. obtained with the validation set by the GMR network trained with data projected in a three-dimensional space.

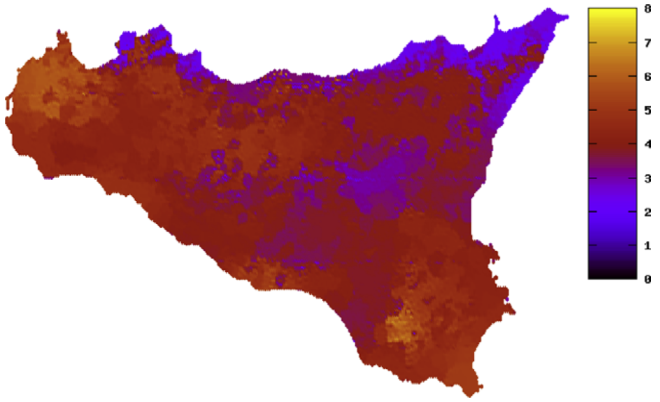

Fig. 13. Map of the wind speed at 50 m a.g.l. obtained starting from the map shown in Fig. 12.

$$\alpha = \frac{0.37 - 0.088\ln\left(U_{z_{ref}}\right)}{1 - 0.088\ln\left(\frac{z_{ref}}{10}\right)} \tag{3}$$

where U is given in m/s and z_{ref} is given in m.

Unlike other expressions that can be used for the estimation of α, such as those by Counihan [24] and by Spera [25], Eq. (3) does not depend upon the *surface roughness length*.

The resulting map of wind speed at 50 m a.g.l. is shown in Fig. 13. In this case, the values of the wind speed attained are higher than those shown in the analogous map contained in the paper by Cellura et al. [8] and closer to the ones contained in the Italian Wind Atlas [26], which were determined using the mass-consistent code wind-field interpolation by non-divergent schemes (WINDS) [27].

5. Discussion of the results

In order to assess the performances of the implemented models, we compared the values of the wind speed at 10 m a.g.l. for the 29 anemometric station locations that were predicted by the two GMR models to those obtained with the NNRK approach [8].

In general, the performance of a neural network must be assessed using data different than those used to train the model; however, in this case, it makes sense to compare the observed and the predicted values of the mean wind speed at the same locations of the anemometric stations. In fact, the two GMR models were not trained directly using the expected values of the Weibull distributions related to the stations, but instead were trained using an artificial TS (see Section 3.3). For each model, we computed the value of the overall absolute percentage error (*APE*), defined as follows:

$$APE = \frac{\sum_{i=1}^{29} \left| \frac{w_i - \mu_i}{\mu_i} \cdot 100 \right|}{29} \tag{4}$$

where w_i is the value of the wind speed estimated by the model for the i-th station and μ_i is the expected value of the Weibull distribution related to the same station.

The APE values obtained with the GMR model trained with data projected on a four-dimensional and a three-dimensional space are 6.52% and 4.70%, respectively. The value obtained with the NNRK model is 9.54%. The maximum and minimum percentage errors obtained with the GMR model trained with data projected on a three-dimensional space are 39.9% (Cammarata station) and 0.017% (Mazara station), respectively. The maximum and minimum percentage errors obtained with the GMR model trained with data projected on a four-dimensional space are 34.0% (Cammarata station) and 0.015% (Canicattì station), respectively. A further test was performed using the data recorded at the Fiumedinisi station. By using the ML method, the shape and scale parameters of the Weibull probability distribution function (*pdf*) fitted to the data were estimated, and the expected value of this distribution was computed. Table 1 shows the values of the mean wind speed estimated by the three models and the related distance from the desired value (i.e., the expected value of the Weibull distribution,

Table 1
Results obtained for the station of Fiumedinisi by GMR trained with data projected in a three-dimensional space (GMR Proj 3), GMR trained with data projected in a four-dimensional space (GMR Proj 4) and the NNRK algorithm.

	GMR Proj 4	GMR Proj 3	NNRK
Estimated mean wind speed (m/s)	2.57	2.67	4.04
Difference from the desired value (m/s)	−1.02	−0.92	0.45
Absolute percentage error (%)	28.41	25.63	12.53

$\mu = 3.59$) in terms of the *APE*. In this case, the NNRK algorithm outperforms the algorithms based on GMR, but this can be justified by the consideration that the Fiumedinisi station is external to the area covered by the stations whose data were used to train the models. For extrapolation problems, *kriging* algorithms perform better than GMR, which is mainly a non-linear function interpolator.

Fig. 14 shows a comparison of the values of the mean wind speed at 50 m a.g.l. In the same graph, the wind speeds found in the Italian Wind Atlas are also reported. The slight underestimation that is still present for the values in the Wind Atlas may be attributed not only to the two completely different approaches used, but also to the different data sources. For this reason, it is not useful to compare the percent error from our models with that made by the Wind Atlas [26].

A comparison with the other neural models employed in the literature to address the same problem is difficult, because the use of neural models applied to the yearly mean wind speed forecasting over complex terrains is new, and little literature exists for comparison. Some applications concerning wind speed forecasting were already discussed in the paper by Cellura et al. [8]; however, they are mostly focused on flat regions and, moreover, are based on considerably longer wind time series and/or more weather stations.

6. Conclusions

A novel neural approach has been used in this paper to tackle the problem of wind speed spatial estimation over the territory of the isle of Sicily (Italy), starting from the measurements recorded at a set of anemometric stations. This is a very difficult task because of the relative shortness of the wind time-series used for the anemometric characterization of each site and the scarcity of measurement stations with regard to the complex topography of the terrain surface. It represents the first application of this neural model to this kind of problem, and is the result of the synergy between a careful exploratory data analysis and the exploitation of the capability of the neural networks to extract the knowledge directly from the data. The core of the data pre-processing phase is represented by the application of the CCA technique for the estimation of the intrinsic dimension of the "wind manifold". The final result of the work is a map of the estimated average wind speed at 50 m a.g.l. The most original aspect of the work is the utilization of a novel neural network architecture and the original way in which information about a variable was inferred from information known about other variables with a higher level of detail. The inference was also made without any assumption about the reciprocal relations between the well-characterized variables and the relatively-unknown variable.

The presented approach is applicable, by following the same procedure outlined here, in any geographical context, provided that enough information about the features of the territory (topography, land coverage, and any other additional items) is known. The MATLAB code for the application of the GMR model can be provided by the authors on request.

However, it must be emphasized that, in general, the results of any mathematical model are to be taken only as a guide for choosing the site of wind farm operations, and even small amounts of

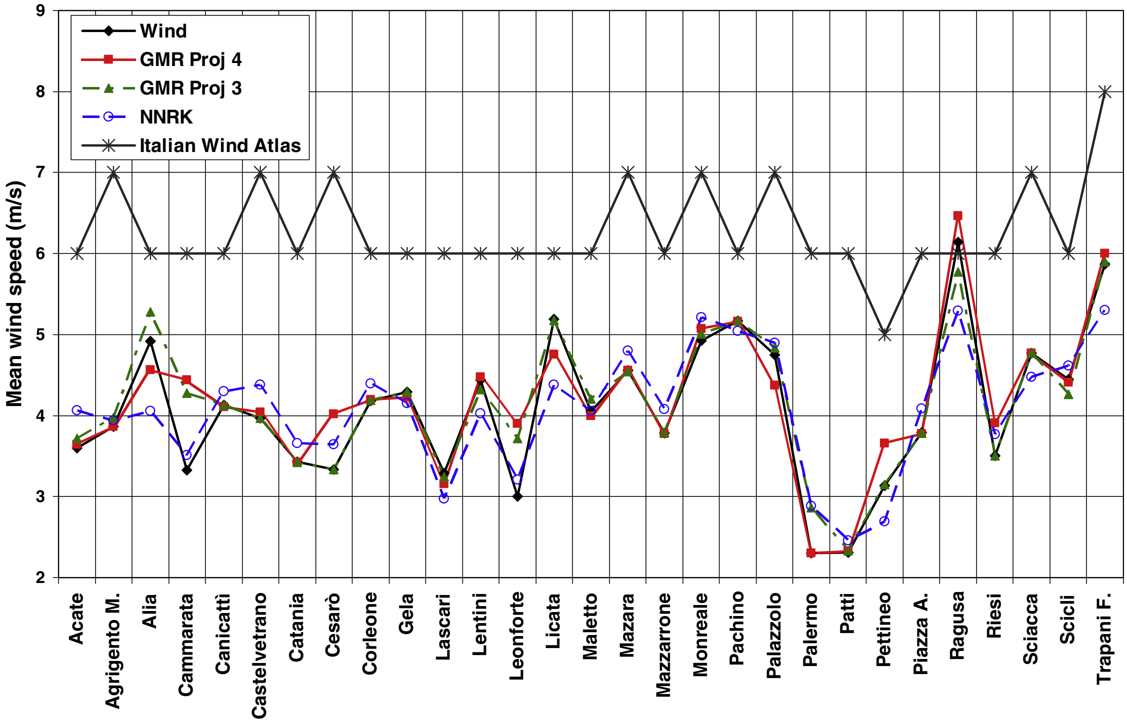

Fig. 14. Comparison of the values of wind speed at 50 m a.g.l. in the locations of the anemometric stations obtained starting from the predictions accomplished by using GMR trained with data projected in a three-dimensional space (GMR Proj 3) and in a four-dimensional space (GMR Proj 4). In the graph are also shown the predictions obtained with the NNRK algorithm [8], the expected values of the Wind distributions related to each anemometric station (Wind) and the wind speed values read in the Italian Wind Atlas [26].

monitoring of the selected areas is always advisable before starting economic investments.

References

[1] Ayotte KW, Davy RJ, Coppin PA. A simple temporal and spatial analysis of flow in complex terrain in the context of wind energy modelling. Bound Layer Meteorol 2001;98:275–95.

[2] Landberg L, Myllerup L, Rathmann O, Lundtang Petersen E, Hoffmann Jørgensen B, Badger J, et al. Wind resource estimation – an overview. Wind Energy 2003;6(3):261–71.

[3] Nielsen M, Landberg L, Mortensen NG, Barthelmie RJ, Joensen A. Application of the measure-correlate-predict approach for wind resource assessment. In: Proc. of the 2001 European wind energy conference and exhibition, Copenhagen (DK) 2–6 July 2001; 2001. p. 773–6.

[4] Landberg L. A mathematical look at a physical power prediction model. Wind Energy 1998;1(1):23–8.

[5] Lalas DP. Wind energy estimation and siting in complex terrain. Int J Sol Energy 1985;3:43–71.

[6] Ratto CF, Festa R, Romeo C, Frumento OA, Galluzzi M. Mass-consistent models for wind fields over complex terrain: the state of the art. Environ Softw 1994;9:247–68.

[7] Dinar N. Mass consistent models for wind distribution in complex terrain. Fast algorithms for three dimensional problems. Bound Layer Meteorol 1984;30:177–99.

[8] Cellura M, Cirrincione G, Marvuglia A, Miraoui A. Wind speed spatial estimation for energy planning in sicily: a neural *kriging* application. Renew Energy 2008;33:1251–66.

[9] Demartine P, Hérault J. Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets. IEEE Trans Neural Networks 1997;8(1):148–54.

[10] Gersho A, Gray RM. Vector quantization and signal compression. London: Kluwer Academic Publishers; 1992.

[11] Hérault J, Jaussions-Picaud C, Guérin-Dugué A, Curvilinear component analysis for high dimensional data representation: I. Theoretical aspects and practical use in the presence of noise. In: Mira J, Sánchez JV, editiors, Proceedings of IWANN'99, vol. II, Alicante, Spain: Springer; 1999. p. 635–44.

[12] Cirrincione G, Cirrincione M, Lu C, Van Huffel S. A novel neural approach to inverse problems with discontinuities (the GMR neural network). In: Proc. of 2003 int. joint conf. on neural networks (IJCNN'03), Portland, Oregon; 2003. p. 3106–11.

[13] Cirrincione G. Neural structure from motion. Unpublished Ph.D. Thesis, LIS INPG, Grenoble, France; 1998.

[14] Cirrincione G, Cirrincione M, Van Huffel S. Mapping approximation by the GMR neural network. In: Proc. of the 4th Wolrd multiconference CSCC 2000, Athens, Greece; 2000, p. 1811–8.

[15] Van Hulle M. Faithful representations and topographic maps: from distortion-to information-based self-organization. New York: Wiley & Sons; 2000.

[16] Davies DL, Bouldin DW. A cluster separation measure. IEEE Trans Pattern Anal Machine Intel 1979;2:224–7.

[17] McQueen J. Some methods for classification and analysis of multivariate observation. In: Proc. of the 5th Berkeley symposium on Math. Stat. and Probab.; 1967. p. 281–97.

[18] Wieringa J. Updating the Davenport terrain roughness classification. J Wind Eng Indus Aerodyn 1992;41:357–68.

[19] Cellura M, Cirrincione G, Marvuglia A, Miraoui A. Wind speed spatial estimation for energy planning in Sicily: introduction and statistical analysis. Renew Energy 2008;33(6):1237–50.

[20] Berthold M, Hand DJ. Intelligent data analysis: an introduction. New York: Springer-Verlag; 1999.

[21] Kitadinis PK. Geostatistics. In: Maidment DR, editor. Handbook of Hydrology. New York: MacGraw-Hill; 1993 [Chapter 20].

[22] Manwell JF, McGowan JG, Rogers AL. Wind energy explained: theory, design and application. New York: Wiley & Sons; 2004.

[23] Justus CG. Winds and wind system performance. Philadelphia: Franklin Institute Press; 1978.

[24] Counihan J. Adiabatic atmospheric. boundary layers: a review and analysis of data collected from the period 1880–1972. Atmos Environ 1975;9:871–905.

[25] Spera DA. Wind turbine technology: fundamental concepts of wind turbine engineering. New York: ASME Press; 1994.

[26] CESI & Università degli Studi di Genova – Dipartimento di Fisica. Ricerca di Sistema per il settore elettrico, Progetto ENERIN: Atlante eolico dell'Italia; 2002. <http://www.ricercadisistema.it>.

[27] Ratto CF, Festa R, Nicora O, Mosiello R, Ricci A, Lalas DP, et al. Wind field numerical simulations: a new user-friendly code. In: Palz W, editor, Proc. European community wind energy conf., Madrid (Spain); 1990. p. 130–4.