

One Day Ahead Prediction of Wind Speed Class

Luigi Fortuna and Silvia Nunnari

Dipartimento di Ingegneria Elettrica, Elettronica ed Informatica
 Universita' degli Studi di Catania
 Viale A. Doria, 6,
 95125 Catania, Italy
 Email: luigi.fortuna,silvia.nunnari@dieei.unict.it

Giorgio Guariso

Dipartimento di Elettronica, Informatica e Bioingegneria
 Politecnico di Milano
 Via G. Ponzio, 34/5
 20133 Milano, Italy
 Email: giorgio.guariso@polimi.it

Abstract—This paper deals with the problem of clustering daily wind speed time series based on two features referred to as W_r and H , representing a measure of the relative daily average wind speed and the Hurst exponent, respectively. Daily values of the pairs (W_r, H) are first classified by means of the fuzzy c-means unsupervised clustering algorithm and then results are used to train a supervised MLP neural network classifier. It is shown that associating to a true wind speed time series a time series of classes, allows performing some useful statistics. Further, the problem of predicting 1-step ahead the class of daily wind speed is addressed by introducing NAR sigmoidal neural models into the classification process. The performance of the prediction model is finally assessed.

Index Terms—wind speed, time series clustering, fcm algorithm, MLP classifier, NAR models.

I. INTRODUCTION

The problem of wind speed time series prediction has been addressed with a large variety of techniques ranging from the traditional Box-Jenkins *ARMA* and *ARIMA* methods, to several kinds of Neural Network based approaches. A review of the young history of methods for short term prediction of such a kind of time series was given in [1], while more recent references can be found in [2] and [3]. Unfortunately, literature results seem pointing out that even by using sophisticated modeling techniques, wind speed time series can be predicted with some accuracy only at short time horizon, i.e. within a few hours. For this reason, the availability of alternative analysis and modeling techniques, such as those that refer to data mining and machine learning, may play a significant role. Indeed to alleviate the problem of predicting future values assumed by a given time series, one might think to associate a series of classes and then trying to predict the class. Time series clustering approaches can be organized into three major categories, depending upon whether they work directly with raw data, indirectly with features extracted from the raw data, or indirectly with models built from the raw data. A survey about time series clustering approaches can be found in [4], while others most recent references can be found in [5]. In the specific field of wind speed time series, some other applications of such a kind of techniques have been proposed for instance by [6], who suggested to classify wind speed time series according to their intensity and consider the Markov chains for their modeling. Decision trees based on *if-then* rules, which are one of the most popular methods

used in machine learning for classification, have been proposed by [7] with the aim of implementing short term wind speed prediction models. In related renewable energy source fields, such as solar radiation time series, classification of daily time series has been proposed by [8]. Data mining techniques and clustering approaches to classify wind speed data in different cities of Turkey have been adopted by [9]. This paper proposes to cluster daily wind speed time series based on two features referred to as W_r , H respectively, which will be introduced in section II. The data set considered for the case study presented in this work was recorded by at the meteorological station located in Como (Italy) and managed by the Politecnico di Milano (Como Campus), during three years, from 2011 to 2013. Como is a town located at about 200 m a.s.l., on the North board of the Po Valley, close to pre-Alps.

II. TWO FEATURES OF DAILY WIND SPEED TIME SERIES

Wind speed time series sampled at a frequency of minutes, as usual occurs, exhibits a clear daily component, which can be pointed out by computing the power spectra density. Such a component is usually referred to as a breeze and can be considered as the effect of solar radiation on wind speed. In time domain such a daily component can be computed by expression (1), since the described averaging process erase the random phenomena (such as clouds effects), leaving the deterministic part of the signal only.

$$W_{dc}(\tau) = \frac{1}{(d_f - d_i + 1)} \sum_{d=d_i}^{d=d_f} D(d, \tau), \quad (1)$$

$$\tau = 1, \dots, N, d = 1, 2, \dots, 365, d_f \geq d_i,$$

The meaning of symbols in expression (1) is the following:

- $W_{dc}(\tau)$, is the wind speed daily component,
- τ is an integer time index, expressing the sample within a generic Julian day d ,
- $D(d, \tau)$ is a matrix, of appropriate dimensions, where wind speed samples recorded at day d and time τ are stored,
- d_i ed d_f represent the initial and final Julian day index which characterize the averaging window.

Examples of wind speed daily patterns in different months of the year, obtained applying expression (1) and assuming τ equal to 1 hour, are shown in Figure (1). These patterns

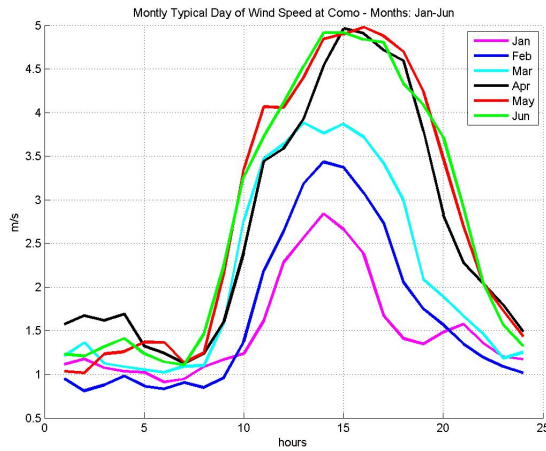


Fig. 1. Examples of daily wind speed components computed averaging on monthly base.

suggest that gradual changes in height and width of the wind speed daily component occur through different months of the year. It can be conjectured that such changes, similarly to what happens for the solar radiation, occur also at daily scale, since they are the effects of Earth's orbiting around the sun. A strategy to compute the solar radiation daily component at daily grain, by using expression (1), was proposed in [10] and can be easily extended to wind speed. The area under the wind speed daily component allows to introduce a feature, referred to as W_r , which represents the basis for the classification of daily wind speed time series proposed in this paper.

A. The area ratio W_r index

The W_r index is a feature of true daily wind speed time series, formally defined as in expression (2)

$$W_r(d) = \frac{A_{true}(d)}{A_{dc}(d)} \quad (2)$$

where:

- $A_{true}(d)$ is the area under the true wind speed time series, recorded at the generic day d ,
- $A_{dc}(d)$ is the area under the wind speed daily component at day d .

Thus, the W_r index, expresses a relative measure of the daily wind speed intensity. An example of W_r daily time series is shown in Figure (2).

B. The Hurst exponent of daily wind speed

In order to perform the classification process, in addition to the W_r index, it was thought to consider a second index that synthesizes the correlation properties of daily wind speed time series. To this purpose, among several possible candidates, it was decided to consider the Hurst exponent, here referred to as H , which can be efficiently computed by several approaches [11]. Values $H < 0.5$ of the Hurst exponent means that the time series is anti-correlated, $H = 0.5$ means that the time series is uncorrelated, i.e. a white noise, while $0.5 < H < 1$

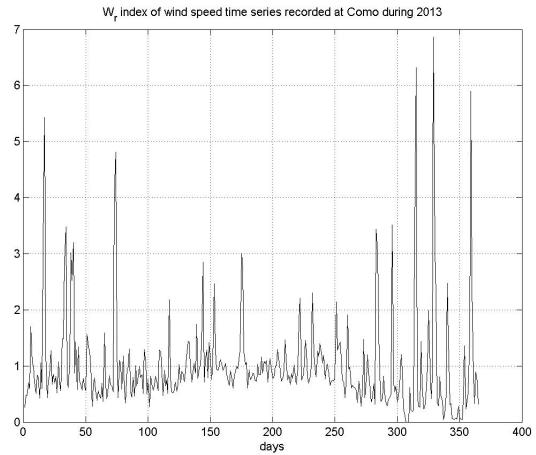


Fig. 2. W_r index computed at the considered station during one year.

means that the time series is positively correlated. The Hurst exponent computed for the 5-minutes daily wind speed time series recorded at the considered station, which means over time series of 288 values each, is shown in Figure (3). The Figure shows that these series are generally positively correlated, since on average the Hurst exponent is about 0.9. Furthermore, it seems that individual daily time series are time varying and behave as a $1/f$ noise.

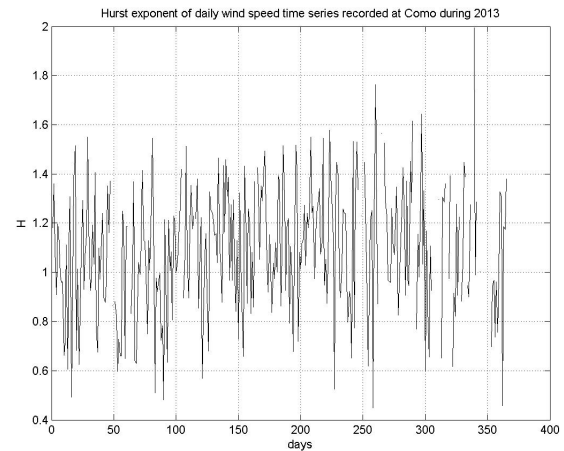


Fig. 3. H index computed at the considered station during one year.

III. WIND SPEED TIME SERIES CLASSIFICATION

The classification approach described in this section consists of the two following steps:

- 1) Daily wind speed time series, each consisting of 288 samples, are mapped into pairs $(W_r(d), H(d))$.
- 2) The pairs $(W_r(d), H(d))$ are clustered into a pre-defined number of classes by using the fuzzy c-means (*fcm*).

The *fcm* algorithm, known in literature since long ago [12], assigns to given set of patterns $\{x_k | k = 1, \dots, n\}$ a predefined

number c of cluster centers, expressed by a set of vectors V , $\{v_i | i = 1, \dots, c\}$, by minimizing an objective function of the form (3)

$$J_m(U, V) = \sum_{i=1}^c \sum_{k=1}^n (\mu_{ik})^m \|x_k - v_i\|^2 \quad (3)$$

where $U = [\mu_{ik}]$ is the fuzzy partition matrix, $\mu_{ik} \in [0, 1], \forall i, k$, represents the degree to which the i_{th} pattern belongs to the k_{th} class, and $1 \leq m \leq \infty$, $m = 2$ being the most popular choice for this parameter. Thus the *fcm* clustering allows a given pattern to belong to different classes but with different degree of membership. However, as done in this application, usually a pattern is assigned to the class with the highest degree of membership. As regards the choice of the number of classes, a parameter required to run the *fcm* algorithm, in this paper different options have been taken into account. Several trials performed have pointed out that the *fcm* algorithm distributes the pairs $(W_r(d), H(d))$ essentially by increasing values of W_r , which thus plays the role of dominant feature. For instance, classification into 3 classes is shown in Figure (4.a) which shows that, roughly speaking class C_1, C_2 and C_3 consist of pattern featured by $0 \leq W_r, 1 \leq W_r \leq 2$, and $W_r > 3$, respectively. For classifications with a number of classes greater than or equal to 4 only, the *fcm* algorithm takes into account the H index as shown, for instance, in the 4-class clustering reported in Figure (4.b). For practical reasons, as done in this paper, once a representative set of patterns has been classified by using the unsupervised *fcm* algorithm, a supervised classifier can be trained in order to classify new incoming patterns.

IV. STATISTICS ON TIME SERIES OF CLASS

Once classes have been attributed to daily wind speed time series, it is possible to perform some statistics working on the time series of class. A useful statistic is that of estimating the persistence, defined as the number of episodes in a year in which a daily pattern persists in the same class for at least p consecutive days. An example of this kind of statistic is given in Figure (5). The Figure refers to a 4-class framework and shows that, at the considered station, classes C_1 and C_2 , i.e. those characterized by low and medium relative wind speed exhibit quite similar behavior and in 2011 about 80 events lasted at least two days, 32 at least 3 days, 18 of at least 4 days and 10 at least 5 days. Furthermore it can be seen that events in class 4, i.e. the class featured by relative high wind speed are not so frequent and only 5 events lasted at least 2 days and only 1 at least 3 days. Thus, no events in class C_4 lasted more than 3 consecutive days. As concerning events in class C_3 , i.e. the class featured by moderately high wind speed, in 2011, 40 events lasted at least 2 days, 14 at least 3 days and only 2 at least 5 days. Another useful statistic is to compute the class weight, i.e. the percentage of daily patterns in each class along one year. This statistic can be easily computed from the class permanence described above. Indeed, it is trivial to observe that indicating as $n_i(p)$ the number of patterns in a year that persist at least p days and as n_c the number of classes, it is

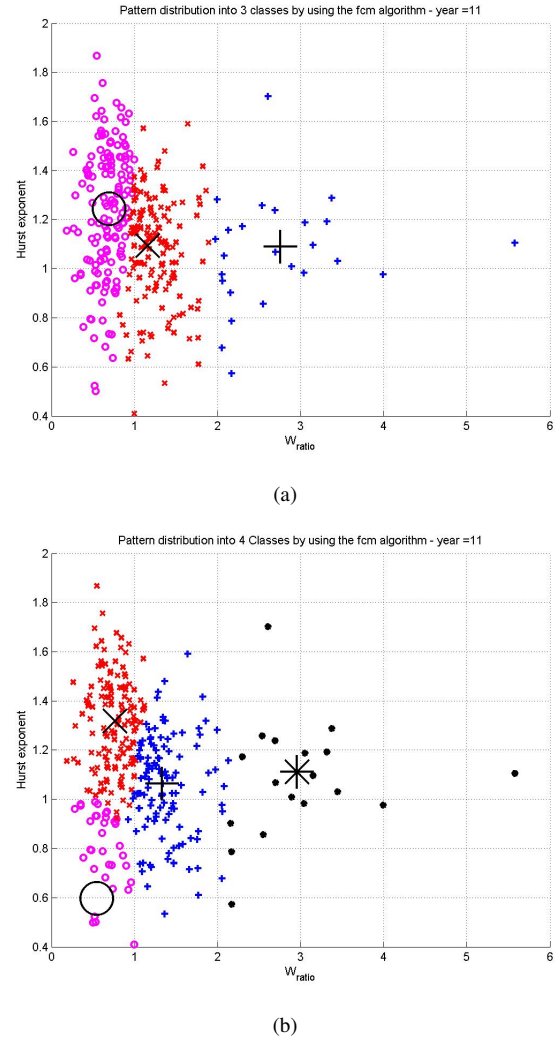


Fig. 4. Classification of pairs (a) 3-class framework (b) 4-class framework.

possible to compute the weight of each class W_i as indicated in expression (4).

$$W_i = \frac{n_i(1)}{\sum_{i=1}^{n_c} n_i(1)} 100 \quad (4)$$

Thus for instance, reading from Figure (5) that $n(1) = 112$, $n(2) = 137$, $n(3) = 94$, and $n(4) = 21$, and applying expression (4) it is possible to obtain that the weights of the 4 considered classes is about 31%, 37%, 26% and 6%, respectively, as shown in Figure (6).

V. PREDICTING THE CLASS

In this section the problem of predicting the class of daily wind speed time series 1-step ahead, will be addressed. The strategy consists of the following two steps:

- 1) identify prediction models for both the features $W_r(d)$ and $H(d)$, in order to compute the predicted pairs $(\hat{W}_r(d)$ and $\hat{H}(d)$),
- 2) use a supervised classifier to associate a class to the predicted pairs $(\hat{W}_r(d), \hat{H}(d))$.

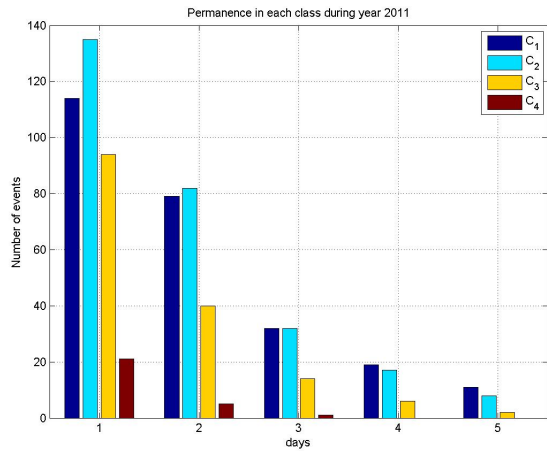


Fig. 5. Permanence in the same class at the considered station.

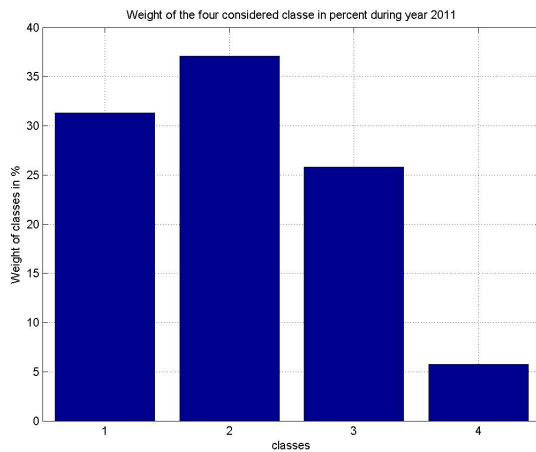


Fig. 6. Weight of classes at the considered station.

The two steps above, performed in sequence makes what in this paper is referred as a 1-day ahead class prediction model.

A. Identify NAR models for $W_r(d)$ and $H(d)$

Dealing with step 1, NAR (Non-linear Auto-Regressive) models were taken into account. It is to bearing in mind that by using NAR models, the future values of a time series $y(t)$ are predicted only from past values, as expressed by (5)

$$y(t) = f(y(t-1), \dots, y(t-n)) \quad (5)$$

where:

- f is an appropriate non-linear function of its arguments $y(t-1), \dots, y(t-n)$, usually referred to as regressors,
- t is the time, that in this application will be evaluated in days,
- n is the number of regressors.

In order to estimate n it may be helpful to evaluate the autocorrelation of the series to be predicted. Since the autocorrelation is a linear feature of time series, which instead are usually

generated by non linear processes, in this paper it was decided to consider the mutual information, defined as in (6)

$$I = - \sum_{i,j} p_{ij}(\tau) \ln \frac{p_{ij}(\tau)}{p_i p_j} \quad (6)$$

where for some partition of the time series range, p_i is the probability to find a time series values in the i_{th} interval and p_{ij} is the joint probability that an observation falls in the i_{th} interval and the observation time τ later falls into the j_{th} interval. The mutual information of the individual $W_r(d)$ and $H(d)$ time series is shown in Figure (7). How it is possible to see, it decays very sharply almost with the same behavior for both the features, reaching the lowest level, for the first time, after about 3 or 4 lags. Based on this result the number of considered delays for the considered application can be set as $n = 4$. Thus this value was assumed in this work for identifying NAR models for both the $W_r(d)$ and $H(d)$ daily time series. As concerning the problem of estimating the non-

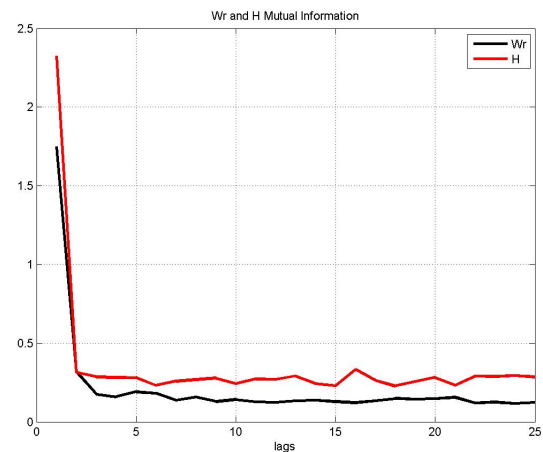
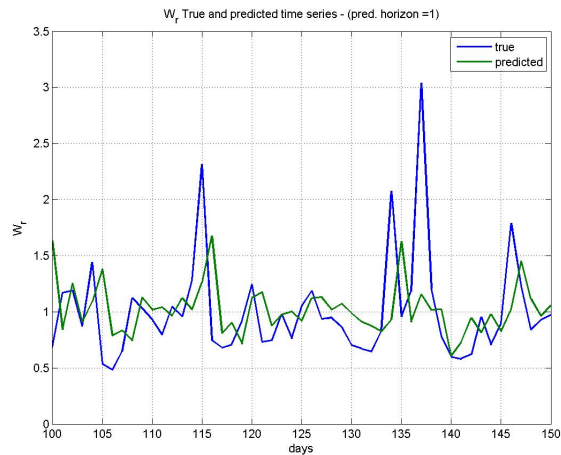
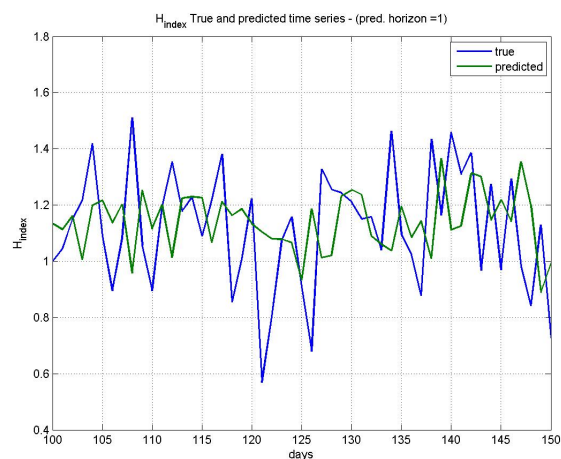


Fig. 7. Mutual information of W_r and H daily values.

linear function f in expression (5), sigmoidal neural networks were considered for all trials shown in this paper. The available data set was divided, as usual in good practice, into a training and a testing data set, in order to avoid polarization of the identified models.

From a visual standpoint, the performances of two independent 1-step ahead neural networks based NAR models, to predict W_r and H daily values, respectively, are shown in Figure (8). As it is possible to see, the prediction models behave quite similar to a persistent model, that is, the predicted time series exhibits peaks delayed by 1-step with respect to the true time series. Furthermore, it seems that models tend to producing somewhat lower peaks. This obviously means that the kind of considered models will have some difficulty to predict patterns featured by high values of W_r . However, it is to bearing in mind that in the considered application, the predicted pairs $\hat{W}_r(d)$ and $\hat{H}(d)$ are not the final target, since they are the input for a MLP classifier, which was appropriately trained to associate a class to each predicted

(a) W_r daily values(b) H daily valuesFig. 8. 1-step ahead prediction of W_r and H by using a NAR model.

pair. Thus, in the final instance, the model evaluation must be performed by comparing the true and the predicted time series of class. A strategy to objectively perform such a comparison and results obtained are described in the next section.

B. Assessing the performance

The problem to objectively comparing a true and a predicted series of classes can be tackled in several ways. In this paper we chose to assess the performance of the classifier in terms of:

- True Positive Rate (TPR), i.e the pattern proportion in a given class which are correctly classified,
- True Negative Rate (TNR), i.e the proportion of patterns which are correctly identified as not belonging to a given class.

Since the TPR and the TNR are also referred to as Sensitivity and Specificity, these terms will be used in the rest of this paper. It is worth nothing that a good predictor would be characterized by values of Sensitivity and Specificity both

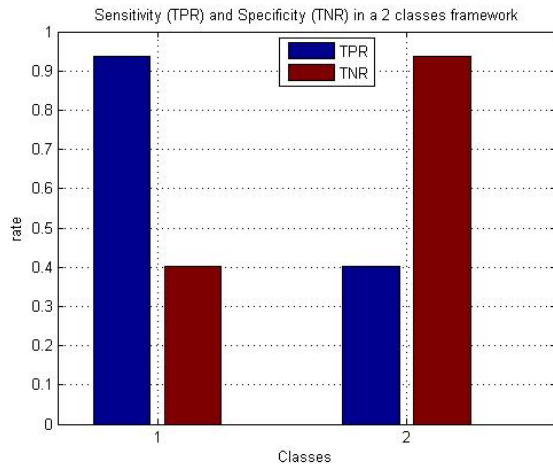
close to 1. Results obtained, for the classification frameworks into 2, 3 and 4 classes, are reported in Figure (9). In more detail, Figure (9.a) shows that in a 2-class framework, there is a high Sensitivity in respect of class C_1 which, however, is featured by a low Specificity. Here it is to bear in mind that for the 2-class framework relations (7) hold:

$$\begin{aligned} TNR(1) &= TPR(2) \\ TNR(2) &= TPR(1) \end{aligned} \quad (7)$$

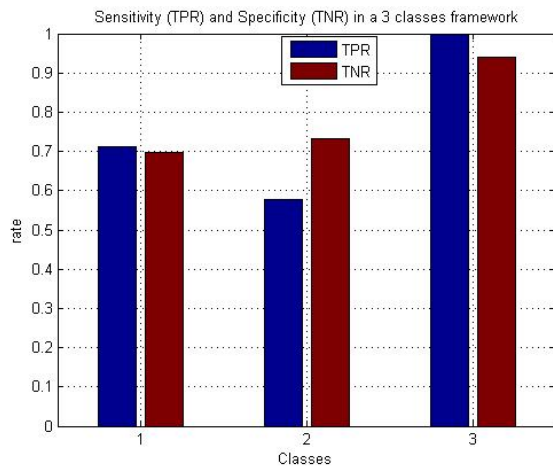
where the integer number in brackets represents the class. Thus in this framework, a low Specificity for class C_1 implies a low Sensitivity for class C_2 . Sensitivity and Specificity for the 3-class framework are shown in Figure (9.b). It is possible to see that in this framework each class is characterized by almost balanced values of Sensitivity and Specificity. Finally, Figure (9.c) shows that in a 4-classes framework, classes C_1 , C_2 and C_3 are characterized by Sensitivity lower than 0.5 thus meaning a poor capability of the classifier to predict patterns belonging to these classes. In summary, this analysis shows that the 3-class framework should be preferred among those considered. Further details about the performances of the 3-class framework are reported in Figure (10). It is possible to see that, in the considered case study, among the 128 + 56 patterns belonging to the target class C_1 , in the testing year, 128 were correctly classified 1-day ahead, while the remaining 56 were attribute to the class C_2 , thus the percentage of events correctly predicted is 69.6%. Similarly the percentage of patterns correctly predicted for the remaining classes C_2 and C_3 are 67.3% and 25.0%. Thus, while the percentage of events correctly classified for classes C_1 and C_2 can be considered satisfactorily, this is not true for the class C_3 . Nevertheless, it should be bear in mind that the events in this class are less frequent than others (only 28 pattern out of a total of 365) and that patterns correctly recognized in this class are almost certain, since the Sensitivity for this class approaches 1.

VI. CONCLUSIONS

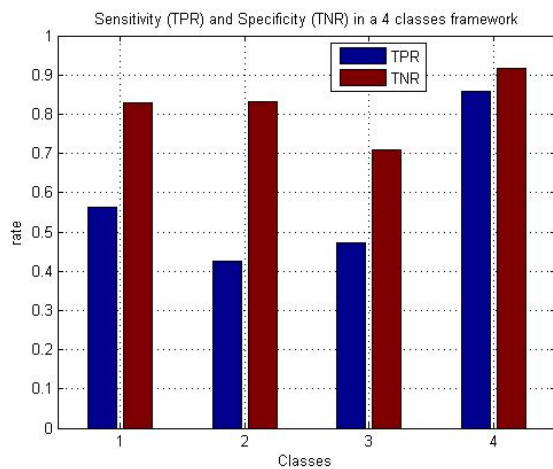
In this paper, a strategy to classify daily wind speed time series, based on a couple of indices referred to as W_r and H , has been proposed. Results show that classes distributes essentially by increasing values of W_r , while the H index seems to play a role when the number of classes is at least 4. Associating to a true wind speed time series a time series of classes, allows performing some useful statistics, such as estimating the weight of each class during a prefixed time interval (for instance one year) and/or the permanence of patterns in each class, which can give some useful insights to managers of wind power plants. Further, the time series of classes allows making 1-day ahead class prediction, which in principle should be less difficult than predicting 1-day ahead the daily average wind speed value. Results obtained so far by using NAR neural network models, with sigmoidal activation functions, shows that clustering wind speed time series into 3 classes is best in terms of Sensitivity and Specificity of the classifier. Further developments are in progress to try



(a)



(b)



(c)

Fig. 9. Sensitivity and Specificity rates (a) 2-class framework (b) 3-class framework (c) 4-class framework.

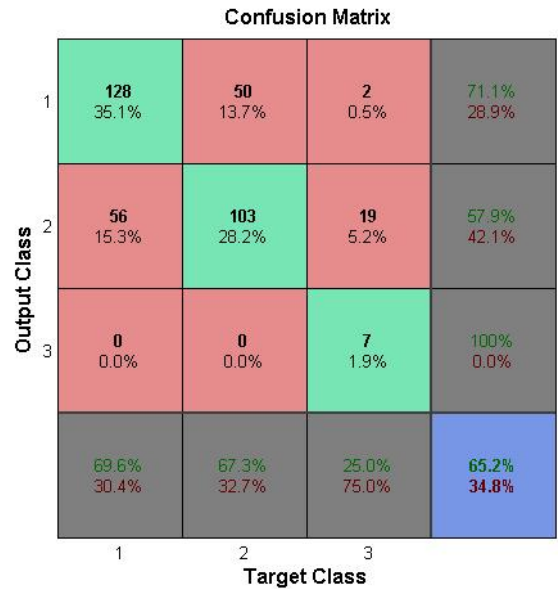


Fig. 10. Confusion matrix of the 1-step ahead model class prediction, for the 3 classes framework.

improving the performance of the class prediction model, mainly for classes featured by high W_r .

REFERENCES

- [1] A. Costa, A. Crespo, J. Navarro, G. Lizcano, H. Madsen, E. Feitosa, A review on the young history of the wind power short-term prediction, *Renewable and Sustainable Energy Reviews* 12 (2008) 1725–1744.
- [2] G. Giebel, R. Brownsword, G. Kariniotakis, M. Denhard, C. Draxl, The state of the art in short-term prediction of wind power - a literature overview, Project ANEMOS Deliverable Report D1.1, 2003 <http://www.anemos-plus.eu/> (2011) 1–110.
- [3] J. Jung, R. P. Broadwater, Current status and future advances for windspeed and powerforecasting 31 (2013) 762-777.
- [4] T. W. Liao, Clustering of time series data - a survey, *Pattern Recognition* 38 (2005) 1857–1874.
- [5] S. Rani, G. Sikka, Recent techniques of clustering of time series data: A survey, *International Journal of Computer Applications* 52 (2012) 1–9.
- [6] Z. Song, X. Geng, A. Kusiak, C. Xu, Mining markov chain transition matrix from wind speed time series data, *Expert Systems with Applications* 38 (2011) 10229-10239.
- [7] A. Troncoso, S. Salcedo-Sanz, C. Casanova-Mateo, J. R. Prieto, Local models-based regression trees for very short-term wind speed prediction, *Renewable Energy* 81 (2015) 589–598.
- [8] T. Soubdhan, R. Emilion, R. Calif, Classification of daily solar radiation distributions using a mixture of dirichlet distributions, *Solar Energy* 83 (2009) 1056–1063. doi:10.1016/j.chaos.2008.07.020.
- [9] S. I. Colak and, M. Demirtas, M. Yesilbudak, A data mining approach: Analyzing wind speed and insolation period data in Turkey for installations of wind and solar power plants, *Energy Conversion and Management* 65 (2013) 185-197.
- [10] L. Fortuna, S. Nunnari, A. Gallo, A Typical Day Based Approach To Detrend Solar Radiation, Proc. of the MAED14, November 7, 2014, Orlando, Florida, USA, <http://dx.doi.org/10.1145/2661821.2661823>.
- [11] R. Weron, Estimating long range dependence finite sample properties and confidence intervals, *Physica A* 312 (2002) 285–299.
- [12] J. Bezdek, R. Ehrlich, W. Full, fcm: the fuzzy c-means clustering algorithm, *Computers & Geosciences* 10 (1984) 191–203.