

# One Day Ahead Prediction of Wind Speed Class by Statistical Models

Luigi Fortuna\*, Giorgio Guariso\*\*, Silvia Nunnari\*‡

\*Dipartimento di Ingegneria Elettrica, Elettronica ed Informatica, Università degli Studi di Catania, Viale A. Doria, 6, 95125 Catania, Italy

\*\*Dipartimento di Elettronica, Informatica e Bioingegneria, Politecnico di Milano, Via G. Ponzio 34/5, 20133 Milano, Italy

‡Corresponding Author; Silvia Nunnari, Viale A. Doria, 6, Catania, Italy, Tel: +39 095 7382306,

Fax: +39-095-330793; silvia.nunnari@dieei.unict

*Received: 10.05.2016-Accepted: 23.07.2016*

**Abstract-**This paper deals with the clustering of daily wind speed time series based on two features, namely the daily average wind speed and the corresponding degree of fluctuation. Daily values of the feature pairs 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 the class of daily wind speed 1-step ahead is addressed by using both the Hidden Markov Models (HMM) and the Non-linear Auto-Regressive (NAR) approaches. The performances of the considered class prediction models are finally assessed in terms of True Positive rate (TPR) and True Negative rate (TNR), also in comparison with the persistent model.

**Keywords** wind speed; time series clustering; fcm algorithm; HMM models; NAR models

## 1. Introduction

Wind speed is an important renewable energy source and a lot of effort has been devoted to study effective prediction techniques. Several kinds of non-linear approaches, mainly based on the use of artificial neural networks [1,4], fuzzy models [5], Bayesian models [6], Support Vector Machine [7], stochastic differential equations [8] have been widely considered.

Recent reviews on techniques for wind speed time series forecasting can be found in [9,11].

Most of these studies demonstrate that forecasting wind speed time series with some accuracy is possible only for short time horizons, i.e. within a few hours. This means that prediction models at daily scale, based solely on the autocorrelation of the wind speed time series properties are doomed to fail. 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 precisely predicting daily averages of wind speed time series, one might think to map them to a series of classes (clusters) and then try 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 nearly survey about time series clustering approaches can be found in [12], while others most recent references are in [13,14].

In [15], it was suggested to classify wind speed time series according to their intensity and the Markov chain was considered as a modeling approach.

The decision trees approach, based on "if-then" rules, which is a popular method used in machine learning for classification purposes, was proposed by [16] for very short-term wind prediction.

In related renewable energy source fields, such as solar radiation time series, classification of daily time series using a mixture of Dirichlet distribution was proposed by [17].

Data mining techniques and clustering approaches to classify wind speed data in different cities of Turkey have been adopted by [18]. A new approach to very short-term wind speed prediction using k-nearest neighbor classification was proposed by [19]. Finally,

spectral clustering and optimised echo state networks for short term wind speed prediction was proposed by [20].

This paper aims to show that clustering techniques allow to perform some useful statistics at daily scale. To this purpose in Section 2 and 3 we show that wind speed daily patterns can be clustered based on two features, referred to as  $\overline{ws}$  and  $D$  respectively, representing the daily average wind speed and the corresponding degree of fluctuation, respectively. In section 4, we present a statistical analysis of wind speed daily patterns, for an Italian meteorological station. In section 5, despite aware of the difficulties of performing one-day ahead prediction of wind speed, we try to predict the wind speed class by two different techniques, namely the HMM and the NAR approaches. In Section 6, performances of these models are objectively assessed in terms of two indices, the TRP (True Positive Rate) and the TNR (True Negative Rate), which are largely used in order to inter-compare time series of classes. Furthermore, in order to evaluate the interest of using sophisticated models such as the HMMs and the NARs, we have evaluated their performances against the simple persistent model, which is often used as a low-reference model. Finally, conclusions of the work are drawn in Section 7.

The data set considered for the case study presented in this work was recorded every five minutes by a meteorological station, located at about 10 m from the ground in Como (Italy) and managed by the Politecnico di Milano (Como Campus), during years from 2011 to 2013. Como is a town located at about 200 m a.s.l., on the North rim of the Po Valley, close to pre-Alps. The general structure of the approach described in this paper were presented in [21].

**2. Two Features of Wind Speed Time Series**

In order to classify wind speed daily patterns, in this paper we consider two features, namely the average daily value  $\overline{ws}$  and the fractal dimension  $D$ , which are defined by equations (1) and (2), respectively.

$$\overline{ws}(t) = \frac{1}{N} \sum_{j=1}^N ws(j, t) \tag{1}$$

$$D(t) = - \lim_{\epsilon \rightarrow 0} \frac{\log n_{\epsilon}}{\log \epsilon} \tag{2}$$

In expression (1),  $N$  is the number of speed samples  $ws(j, t)$  collected at instant  $j$  of day  $t$ , while in expression (2),  $\epsilon$  is a small square lattice with side  $\epsilon$  and  $n_{\epsilon}$  is the number of grids needed to cover the time series. The rationale for choosing these two features, among several others possible, is that  $\overline{ws}(t)$  can be related to the daily average energy that can be obtained from wind, while  $D(t)$  can be related with the degree of daily energy fluctuation.

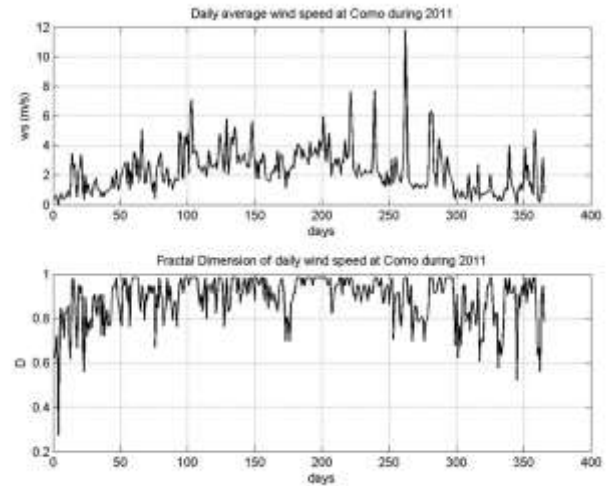
While the calculation of (1) is trivial, estimation of  $D$  for each day  $t$  is based on the assumption that the relation between  $n_{\epsilon}$  and  $\epsilon$  is a power law (3)

$$n_{\epsilon} \propto \epsilon^{-D} \tag{3}$$

Thus, by taking the log of both members in expression (3), we have:

$$\log n_{\epsilon} = \log C - D \cdot \log \epsilon \tag{4}$$

which represents a straight line in a log-log diagram, drawn in the plane  $n_{\epsilon}$  versus  $\epsilon$ . The angular coefficient of this line is  $D$  while  $C$  is a constant. In order to estimate  $D$ , it is then possible to approximate the curve  $\log n_{\epsilon}$  versus  $\log \epsilon$  with a regression line by using the traditional least square approach. The two considered features, computed for the Como station during 2011, are shown in Fig. 1.



**Fig. 1.** Features of wind speed daily patterns.

As it can be immediately seen, these features exhibit a quite irregular behaviour. In view of modelling these features by using NAR models (see section 5.2), it might be helpful to evaluate the autocorrelation of the series. However, since the autocorrelation is a linear feature of time series, which instead are usually generated by non-linear processes, in this paper the mutual information was considered. It is defined as

$$I = - \sum_{i,j} p_{ij}(\tau) \log \frac{p_{ij}(\tau)}{p_i p_j} \tag{5}$$

In this expression, for some partition of the time series range,  $p_i$  is the probability to find a value in the  $i$ -th interval and  $p_{ij}$  is the joint probability that an observation falls in the  $i$ -th interval and the observation at the following time step falls into the  $j$ -th interval. The mutual information of the individual  $\overline{ws}$  and  $D$  time series is shown in Fig. 2.

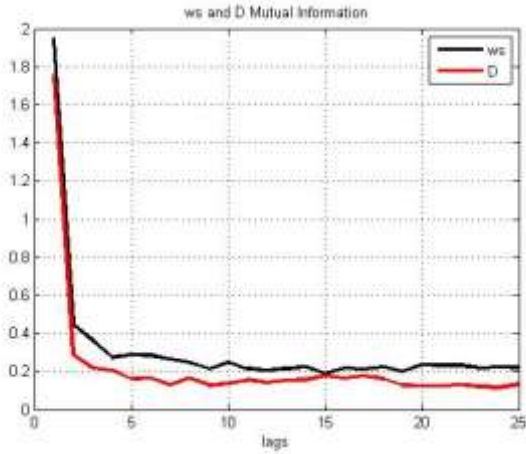


Fig. 2. Mutual information of  $\bar{ws}$  and  $D$  daily samples.

As it is possible to see, the mutual information suddenly decays after just 1 lag, thus implying that one-day ahead prediction of the class is extremely difficult by using autoregressive models. Therefore, clustering approaches can be useful to extract, at least, some statistical information at daily scale.

3. Wind Speed Time Series Clustering

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

- Daily wind speed time series are mapped into pairs ( $\bar{ws}$  and  $D$ ).
- These pairs are then clustered into a pre-defined number of classes by using the fuzzy c-means (*fcm*) algorithm.

The *fcm* algorithm [22] assigns to a given set of patterns  $\{x_k | k = 1, \dots, n\}$  a predefined number  $c$  of cluster centers, represented by a set of vectors  $V, \{v_i | i = 1, \dots, c\}$ , by minimizing an objective function of the form

$$J_m(U, V) = \sum_{i=1}^c \sum_{k=1}^n (\mu_{ik})^m \|x_k - v_i\|^2 \quad (6)$$

In expression (6),  $U = [\mu_{ik}]$  is the fuzzy partition matrix,  $\mu_{ik} \in [0,1] \forall i,k$ , represents the degree at 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 with different degrees of membership. However, as done in this paper, 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, a classification into 3 classes was considered, based on results described in [21].

An example of 3-class clustering of daily wind speed features, computed during 2011, is shown in Fig. 3.

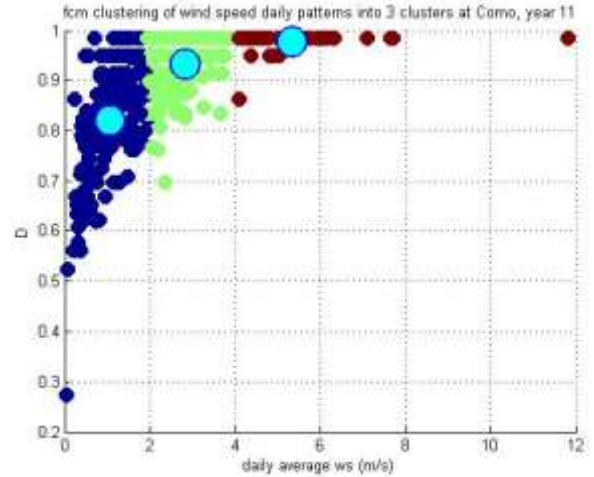


Fig. 3. Clustering the features computed during 2011 into 3 classes; the circles represent the cluster centres.

Fig. 3 shows that daily patterns are essentially distributed over the classes by increasing values  $\bar{ws}$ . Nevertheless, there is a discrimination also in terms of  $D$ , since this feature has narrowing ranges going from class  $C_1$  to class  $C_3$ .

In order to have a rough assessment of the clustering, we computed the corresponding silhouette, as shown in Fig. 4. The silhouette  $S(i)$  is a measure, ranging from -1 to 1, of how well the  $i$ -th pattern lies within its cluster:  $S(i)$  close to 1 means that the corresponding pattern is appropriately clustered; on the contrary, if  $S(i)$  is close to -1, then the  $i$ -th pattern would be more appropriately clustered in its neighbouring cluster; finally  $S(i)$  near zero means that the pattern is on the border of two natural clusters.

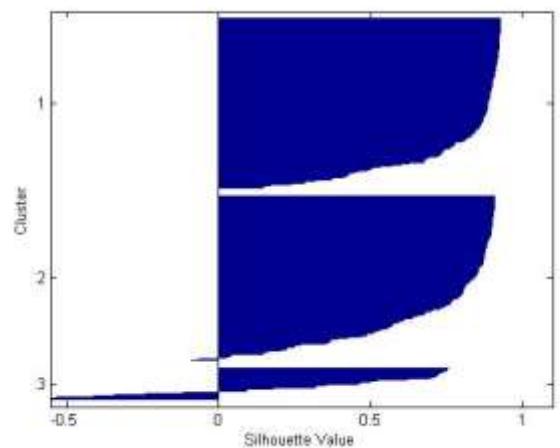
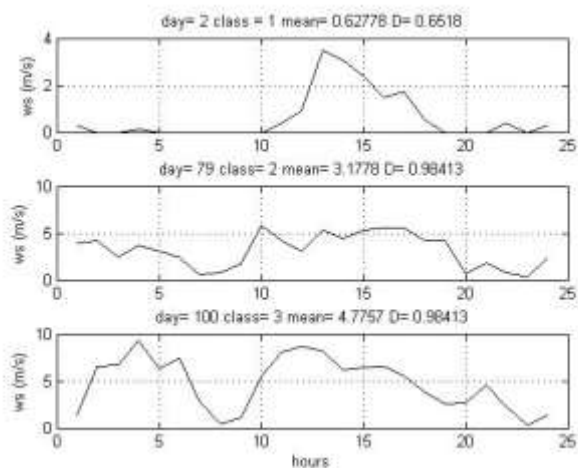


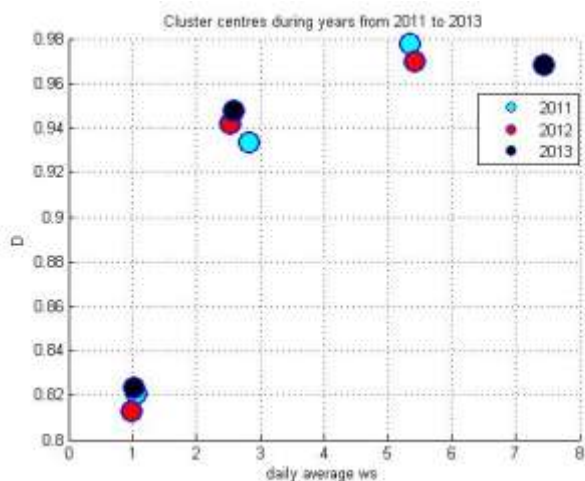
Fig. 4. Silhouette of the clustering shown in Fig. 3.

It is possible to see that while patterns of classes  $C_1$  and  $C_2$  are fairly well clustered, there is a fraction of patterns assigned to the class  $C_3$ , which is the less numerous, that would be better fitted in Class  $C_2$ . Typical patterns of class  $C_1, C_2$  and  $C_3$  are shown in Fig. 5.



**Fig. 5.** Typical wind speed daily patterns recorded at Como: class  $C_1$  (upper), class  $C_2$  (middle), class  $C_3$  (lower).

A further aspect of the clustering performed at the Como station is that the coordinates of the cluster centers are almost time independent, except for a somehow stronger wind in 2013. Indeed, the cluster centers computed yearly, from 2011 to 2013, are shown in Fig. 6.



**Fig. 6.** Wind speed daily patterns cluster centers at Como during 2011-2013; from left to right the centers of class  $C_1$ ,  $C_2$  and  $C_3$ .

The impact of this wind speed change will be evaluated in section 5.1.

#### 4. Some Applications of Clustering Wind Speed Daily Patterns

Once classes have been attributed to daily wind speed time series, it is possible to perform some interesting statistics of the time series of classes.

##### 4.1. Estimating the persistence

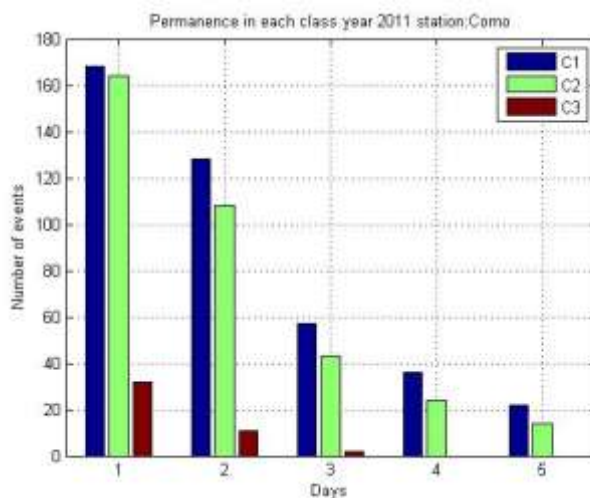
A useful statistic is the estimation of the persistence, defined as the number of episodes in a year in which a daily pattern persists in the same class for at least  $p$  consecutive days. An example of this kind of statistic is given in Fig. 7.

The Figure shows that, during 2011, at the considered station, about 128 events lasted at least two days in class  $C_1$ , 57 at least 3 days, 36 of at least 4 and 22 at least 5 days. A similar behavior holds also for class  $C_2$ . Instead, for class  $C_3$ , i.e. the class characterized by relative high daily average wind speed, persistence is very low: only 11 events lasted at least 2 days and only 2 at least 3 days.

##### 4.2. Estimating the Class Weight

Another interesting 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 in class  $i$  and as  $n_c$  the number of classes, it is possible to compute the weight  $W_i$  of each class as indicated in expression (7).

$$W_i = \frac{n_i(1)}{\sum_{i=1}^{n_c} n_i(1)} 100 \quad (7)$$



**Fig. 7.** Permanence in the same class at the considered station.

Thus for instance, reading from Fig. 6 that  $n_1(1) = 168$ ,  $n_2(1) = 164$ ,  $n_3(1) = 32$ , and applying expression (7) it is possible to obtain that the weight of the 3 considered classes is about 46%, 45%, 9%.

#### 5. Predicting the class

In this section, the problem of predicting the class of daily wind speed time series one-day ahead, will be tackled by using two different approaches, namely the Hidden Markov Model (HMM) and the Non-linear Auto-Regressive

(NAR) models. For all trials described in this paper the overall data set, consisting of three years of data, two years were alternately considered for training the models, while the remaining year was used for testing (thus adopting the classical leave-one-out technique).

5.1. Predicting the Class by using HMM

HMM is a popular statistical tool for modeling a wide range of time series [23]. In more detail, we observe a sequence of emissions (outputs), but we do not know the sequence of states the model went through to generate the emissions. Thus, in general a HMM model is characterized by two matrices, referred to as the transition matrix and the emission matrices, respectively.

The idea underlying the identification of a *h*-step-ahead prediction model by a HMM is that of associating the true time series of classes to the states of the Markov chain and the emissions with the true *h*-step ahead classes.

Thus, in general, a HMM model is characterized by a transition matrix (*T*) and an emission matrix (*E*). However, these two matrices coincide when the prediction horizon *h* = 1, i.e. one-day-ahead.

As an example, in a 3-classes framework, the transition matrix, computed by using the 2011-12 training set is the following:

$$T = \begin{pmatrix} 0.7456 & 0.2251 & 0.0292 \\ 0.2455 & 0.6636 & 0.0909 \\ 0.1034 & 0.5862 & 0.3103 \end{pmatrix}$$

The value of the generic *i,j* entry of this matrix represents the probability that a pattern belonging to the *i*-th class at time *t* transits to the *j*-th class at time *t+1*. Of course, the sum of probabilities in each row is equal to 1, while the values on the diagonal of this matrix represent the probability that a pattern in class *C<sub>i</sub>*, *i* = 1,2,3, remains in the same class for the next day. Transition and Output matrices can be computed by using available algorithms, such as the *hmmestimate* function implemented in the Matlab® statistical toolbox. Nevertheless, it could be interesting to observe that the entries of the *T* matrix can be derived from the permanence analysis described in section 4.1. Finally, for the purpose of predicting the class, the output matrix must be transformed into the most probable state path by using one of the available algorithms, such as the popular Viterbi one [24].

The performances of a one-day ahead HMM class prediction model obtained by using the leave-one-out approach are reported in Fig. 8 in terms of patterns correctly predicted in each testing year.

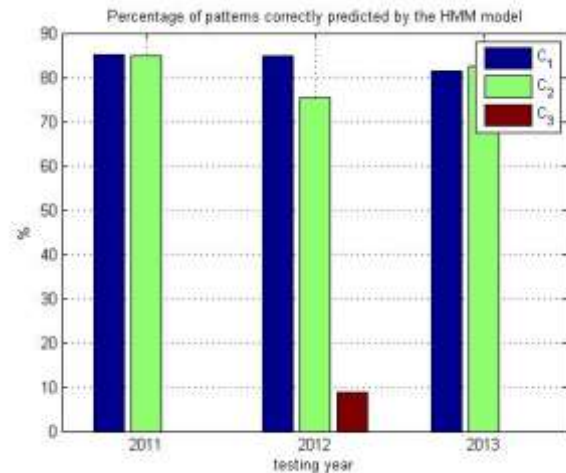


Fig. 8. Percentage of patterns correctly predicted by the HMM model.

Figure (8) shows that the percentage of patterns correctly predicted for the classes *C<sub>1</sub>* and *C<sub>2</sub>* is always better than 75%, while the model is usually not able to predict a significant percentage of patterns in class *C<sub>3</sub>*. Averaging over the whole data set, the percentage of patterns correctly predicted was 77.5%, 75.7% and 77.3% for the testing years 2011, 2012 and 2013, respectively, which is almost constant in time. We will further discuss these results in section 6, in comparison with the performances of the NAR and the persistent models, also in terms of True Positive Rate (TPR) and True Negative Rate (TNR).

5.2. Predicting the Class by using NAR Models

By using NAR models [25], the future values *y(t+1)* of a time series are predicted only from values till day *t*, as in (8)

$$y(t + 1) = f(y(t), y(t - 1), \dots, y(t - n)) \quad (8)$$

where:

- *f* is an appropriate non-linear function of its arguments *y(t), ..., y(t - n)*, usually referred to as regressors,
- *n* is the maximum delay and defines the number of regressors.

In order to estimate *n*, it may be helpful to evaluate the mutual information of the series to be predicted. Based on result shown in section 2, *n* = 4 was assumed, since the mutual information reaches its lowest value approximately after 4 lags. As to the problem of estimating the nonlinear function *f* in expression (8), sigmoidal neural networks were considered for all trials shown in this paper. Predicting the time series of class by using NAR models thus consists of the following two steps:

1. identify NAR models for both the features  $\overline{ws}(t)$  and *D(t)*, in order to compute the one-day ahead predicted pair  $(\widehat{ws}(t + 1), \widehat{D}(t + 1))$

- 2. associate a class to the predicted pair  $(\widehat{ws}(t + 1), \widehat{D}(t + 1))$ .

The two steps above, performed in sequence, make a one-day ahead NAR class prediction model. The available data set was divided again into a training and a testing 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  $\overline{ws}(t)$  and  $D(t)$  values, respectively, are shown in Fig. 9 and Fig. 10, respectively.

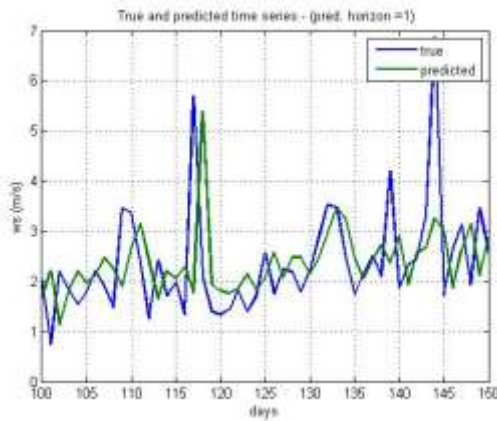


Fig. 9. 1-step ahead prediction of  $\overline{ws}(t)$  by using a NAR model

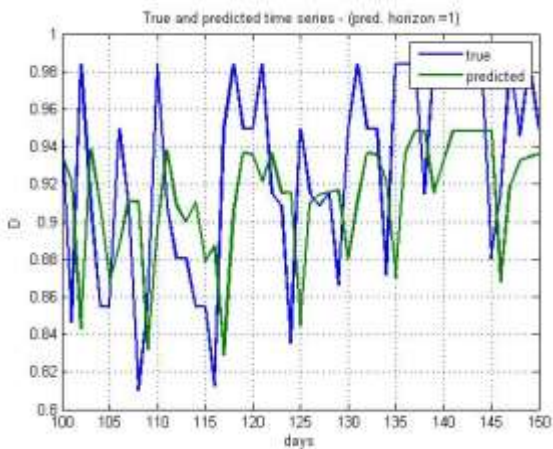


Fig. 10. 1-step ahead prediction  $D(t)$  by using a NAR model

As it is possible to see, these prediction models behave quite similarly to a persistent model, since the predicted time series exhibits peaks delayed of 1-step with respect to the true time series. Furthermore, the predicted peaks are lower than true ones. This means that the considered NAR models will not be able, in general, to predict patterns featured by high values of  $\overline{ws}(t)$  and  $D(t)$ . However, it is to bear in mind that, in the considered application, the predicted pairs  $\overline{ws}(t)$  and  $D(t)$  are not the final target, since they are the input of a MLP classifier, which is trained to associate a class to each predicted pair. Thus, in the final instance, the model evaluation must be performed by comparing the true and the

predicted time series of class. The performances of such a model, computed by using the leave-one-out approach, is shown in Fig. 11.

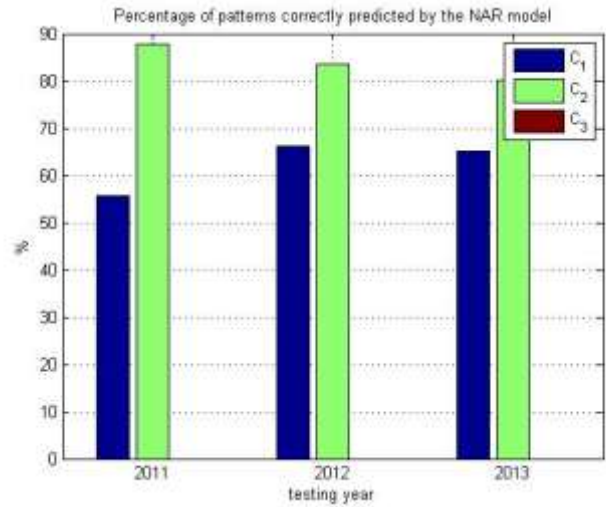


Fig. 11. Percentage of patterns correctly predicted by the NAR model.

It is possible to see that the percentage of patterns correctly predicted is usually better than 80% for the class  $C_2$ , while for the class  $C_1$  is in the range from about 55% (year 2011) to about 66% (years 2012 and 2013). Furthermore, this model was not able to correctly predict any pattern of the class  $C_3$ . Averaging over the whole data set, the percentage of patterns correctly predicted was 65.2%, 70.2% and 68.2% for the testing years 2011, 2012 and 2013, respectively.

### 5.3. Predicting the Class by using the Persistent Model

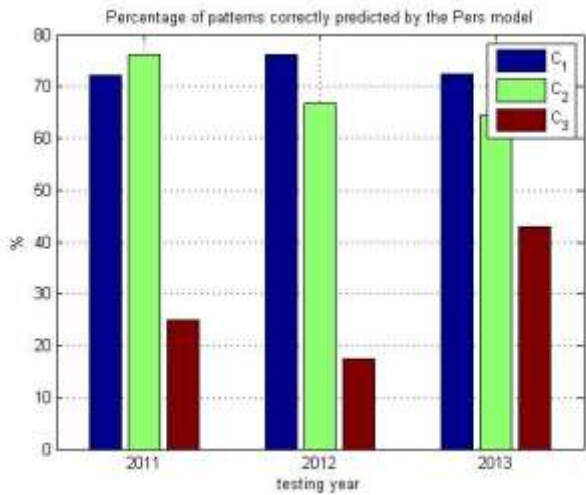
The persistent model is a very simple model, characterized by expression (9), often considered in literature as a low-reference model.

$$\hat{y}(t + h) = y(t) \tag{9}$$

In expression (9),  $y$  is the modeled variable,  $t$  the actual time and  $h$  the prediction horizon. In our application, this model was considered to obtain the one-day ahead prediction of the pair  $\overline{ws}(t)$  and  $D(t)$  by simply setting:

$$\begin{aligned} \widehat{ws}(t + 1) &= \overline{ws}(dt), \\ \widehat{D}(t + 1) &= D(t) \end{aligned} \tag{10}$$

The performances of the described one-day ahead persistent model, computed by the leave-one-out approach is shown in Fig.12. The percentage of patterns correctly predicted is usually better than 70% and 65% for the class  $C_1$  and  $C_2$ , respectively.



**Fig. 12.** Percentage of patterns correctly predicted by the persistent model.

Furthermore, the persistent model is able to correctly predict a limited percentage of patterns in class C<sub>3</sub>, thanks to the persistence effect discussed in section 4.1.

Averaging on the whole data set, the percentage of events correctly predicted by the persistent model is 69.9%, 68.0 % and 67.1% for the testing years 2011, 2012 and 2013, respectively, which is again quite constant in time.

**6. Model Comparison**

In order to inter-compare the predictive performances of the HMM, NAR and persistent models, we consider the two following statistical indices:

- the True Positive Rate (TPR), expressed by (11)

$$TPR(i) = \frac{TP(i)}{P(i)} = \frac{TP(i)}{TP(i) + FN(i)}, \quad i = 1 \dots c \quad (11)$$

where  $TP(i)$  and  $FN(i)$  is the number of true positive and false positive, respectively, attributed by the model to the class  $C_i$ . The sum  $P(i) = TP(i) + FN(i)$  is, of course, the total number of patterns attributed by the model to the class  $C_i$ .

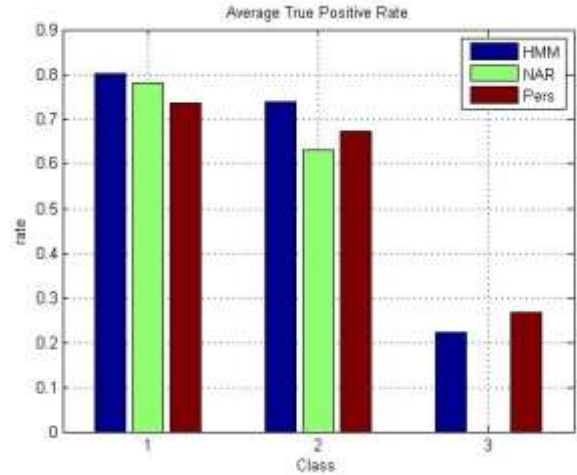
- the True Negative Rate (TNR), expressed by (12)

$$TNR(i) = \frac{TN(i)}{N(i)} = \frac{TN(i)}{TN(i) + FP(i)}, \quad i = 1 \dots c \quad (12)$$

where  $TN(i)$  is the number of patterns which are correctly identified as not belonging to the class  $C_i$  and  $FP(i)$  is the number of false positives attributed by the model to the class  $C_i$ . The sum  $N(i) = TN(i) + FP(i)$  is the total number of patterns recognized by the model as not belonging to the class  $C_i$ .

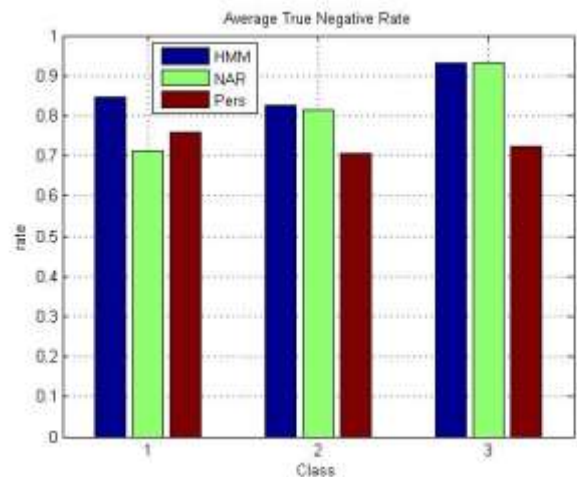
Clearly, a good predictor would be characterized by values of TPR and TNR both close to 1.

Average results over the three testing years are reported in Fig. 13 and Fig. 14, in terms of TPR and TNR, respectively.



**Fig.13.** TPR for a 3-class framework.

They show that, limited to the prediction of the patterns belonging to the classes C<sub>1</sub> and C<sub>2</sub>, the HMM model is the one that exhibits the best performance in terms of TPR and TNR. The main problem of all the considered approaches, is that they are not able to correctly predict a reasonable percentage of patterns belonging to the class C<sub>3</sub>.



**Fig. 14.** TNR rates for a 3-class framework

Indeed, the number of patterns belonging to the class C<sub>3</sub> during one year are, on average, only 25 out 365, and thus with low significance from the statistical point of view. On the other hand, results show that, for the HMM, the rate of correctly classified patterns, averaged over the rotating test set, is better than 0.80 and 0.72 for the class C<sub>1</sub> and C<sub>2</sub>, respectively, which are the classes of greatest weight in the course of the year.

It is also to be stressed that the HMM model exhibits a TNR which is better than 0.8 for all considered classes,

which means that it has a high specificity in recognizing the patterns not belonging to a given class. In summary, it can be said that HMM one-day ahead class prediction models may provide interesting insights to users, such as the managers of wind speed power plants.

## 7. Conclusions

In this paper, a strategy has been proposed to classify daily wind speed time series into three classes, based on two indices: the daily average speed  $\overline{ws}(t)$  and the corresponding degree of fluctuation  $D(t)$ . Results show that classes distribute primarily by increasing values of  $\overline{ws}(t)$ , while  $D(t)$  helps to discriminate among the classes. One of the achievements of the paper is to point out that clustering allows to perform some useful statistics, such as the persistence of patterns in a given class, which, due to the difficulties of making reliable one-day ahead predictions of the average wind speed, may give some useful surrogate information to managers of wind power plants.

Different modelling approaches have been tested in this study to predict one-day ahead the class of wind speed. Despite the known difficulties of predicting relatively high (and hence, rare) wind speed classes, the hidden Markov chain model turned out a bit better from the statistical viewpoint, particularly for the classes  $C_1$  and  $C_2$ , which are the most significant in terms of weights. None of the considered models is indeed able to predict an acceptable number of patterns belonging to the class  $C_3$ . The main reason is that, according to the analysis provided in Section 4, this class is characterized by a low number of patterns (about 9% in terms of weights) and the persistence is very low. Further research is thus needed to understand whether the complexity of the suggested approaches is paid for by their improved statistical performances. Clearly, this problem could be analysed only considering the specific costs and benefits of an actual wind plant and its role in an overall grid.

## References.

[1] C. Ren, N. An, J. Wang, L. Li, B. Hu, D. Shang, Optimal parameters selection for BP neural network based on particle swarm optimization: A case study of wind speed forecasting, *Knowledge-Based Systems*, 56 (2014), pp. 226-239.

[2] Z. Guo, W. Zhao, H. Lu, J. Wang, Multi-step forecasting for wind speed using a modified emd-based artificial neural network model, *Renewable Energy*, 37 (2012), pp. 241-249.

[3] O. B. Shukur, M. H. Lee, Daily wind speed forecasting through hybrid kf-ann model based on arima, *Renewable Energy*, 76 (2015), pp. 637-647.

[4] J. Wang, W. Zhang, J. Wang, T. Han, L. Kong, A novel hybrid approach for wind speed prediction, *Information Sciences*, 273 (2014), pp. 304-318.

[5] B. Zou, M.Y. Chen, N. Wade, L. Ran, A prediction model for wind farm power generation based on fuzzy modeling, *Procedia Environment Science*, 12 (2012), pp. 122-129.

[6] Y. Jiang, Z. Song, A. Kusiak, Very short term wind speed forecasting with Bayesian structural break model, *Renewable Energy*, 50 (2013), pp. 637-647.

[7] J. Zhou, J. Shi, G. Li, Fine tuning support vector machines for short-term wind speed forecasting, *Energy Conversion Management*, 52 (2011), 1990-1998.

[8] E. B. Iversen, J.M. Morales, J.K. Moller, H. Madsen, Short-term probabilistic forecasting of wind speed using stochastic differential equations, *International Journal of Forecasting*, 32 (2016), pp. 981-990.

[9] J. Wang, Y. Song, F. Liu, R. Hou, Analysis and application of forecasting models in wind power integration: A review of multi-step-ahead wind speed forecasting models, *Renewable and Sustainable Energy Reviews*, 60 (2016), pp. 960-981.

[10] A. Tascikaraoglu, M. Uzunoglu, A review of combined approaches for prediction of short-term and power, *Renewable and Sustainable Energy Reviews*, 34 (2014), pp. 243-254.

[11] J. Jung, R. P. Broadwater, Current status and future advances for wind speed and power forecasting, 31 (2013), pp. 762-777.

[12] T. W. Liao, Clustering of time series data - a survey, *Pattern Recognition*, 38 (2005), pp. 1857-1874.

[13] S. Aghabozorgi, A. S. Shirkhorshidi, T. Y. Wah, Time-series clustering - A decade review, *Information Systems*, 53 (2015), pp. 16-38.

[14] S. Rani, G. Sikka, Recent techniques of clustering of time series data: A survey, *International Journal of Computer Applications*, 52 (2012), pp. 1-9.

[15] 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), pp. 10229-10239.

[16] A. Troncoso, S. Salcedo-Sanz, C. Casanova-Mateo, J. R., L. Prieto, Local models-based regression trees for very short-term wind speed prediction, *Renewable Energy*, 81 (2015), pp. 589-598.

[17] T. Soubdhan, R. Emilion, R. Calif, Classification of daily solar radiation distributions using a mixture of dirichlet distributions, *Solar Energy*, 83 (2009), pp. 1056-1063.

[18] S. I. Colak, 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), pp. 185-197.



- [19] M. Yesilbudak, S. Sagiroglu, I. Colak, A new approach to very short term wind speed prediction using k-nearest neighbor classification, 69 (2013), pp. 77-86.
- [20] D. Liu, J. Wang, H. Wang, Short-term wind speed forecasting based on spectral clustering and optimised echo state networks, Renewable Energy 78 (2015), pp. 599-608.
- [21] L. Fortuna, G. Guariso, S. Nunnari, One day ahead prediction of wind speed class, Proc. of the 2015 International Conference on Renewable Energy Research and Applications (ICRERA), Palermo (Italy), Nov. 2015, pp. 965 - 970.
- [22] J. Bezdek, R. Ehrlich, W. Full, fcm: the fuzzy c-means clustering algorithm, Computers & Geosciences, 10 (1984), pp. 191-203.
- [23] L. R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE, 77 (2) (1989), pp. 257-286.
- [24] A.J. Viterbi, Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, IEEE Transactions on Information Theory 13 (2) (1962), pp. 260-269.
- [25] S.A. Billings, Nonlinear System Identification: NARMAX Methods in the Time, Frequency and Spatio-Temporal Domains, Wiley, 2013. (Book)