

Evaluating nominal and ordinal classifiers for wind speed prediction from synoptic pressure patterns

P.A. Gutiérrez[†], S. Salcedo-Sanz[‡], C. Hervás-Martínez[†], L. Carro-Calvo[‡], J. Sánchez-Monedero[†] and L. Prieto^{*}

[†]Department of Computer Science and Numerical Analysis
Universidad de Córdoba

[‡]Department of Signal Theory and Communications
Universidad de Alcalá

^{*}Department of Wind Resource
Iberdrola Renovables

Abstract—This paper evaluates the performance of different classifiers when predicting wind speed from synoptic pressure patterns. The prediction problem has been formulated as a classification problem, where the different classes are associated to four values in an ordinal scale. The problem is relevant for long term wind speed prediction and also for wind speed reconstruction in areas (mainly wind farms) where there are not direct wind measures available. The results obtained in this paper present the Support Vector Machine as the best tested classifier for this task. In addition, the use of the intrinsic ordering information of the problem is shown to improve classifier performance.

Index Terms—ordinal classification, ordinal regression, wind speed, pressure patterns, long-term wind speed prediction, wind farms.

I. INTRODUCTION

Long-term wind speed prediction and wind speed series analysis and reconstruction are important problems in wind farms management. Existing approaches for both problems are mainly based on historic registers of wind measures, from which statistical models are constructed in order to explain the wind behaviour. These models can be then applied to future values of time in the case of long-term wind speed prediction, or to values in the past in order to reconstruct or analyze wind speed series. Different techniques have been used to obtain these wind speed models, such as statistical methods [27], [17], neural networks [21], [1], [20], support vector machines [22], etc. The majority of the existing techniques to construct long-term wind speed models are exclusively based on past wind speed data, and some of them include other atmospheric variables as input data such as local temperature, radiation or pressure at the measuring point. The problem with this approach based on wind measures is that, in some cases, these data are not available, due to problems in the measurement systems, or just because the terrain is a prospective site to install a wind farm, and there is not a meteorological tower installed yet. This problem is even harder in the case of historic analysis or wind series reconstruction, since it is not possible to obtain any direct wind measure if it is not available.

In these cases, the possibility of obtaining indirect measures of wind is currently a hot topic, in which many companies

owning wind farms are investing lots of resources. In the case of the wind, it seems reasonable that a possible source of indirect wind measures is the pressure pattern at synoptic scale, since the wind at a given point is a direct function (when we remove the effects of limit boundary layer) of the pressure gradient. Specifically, in this paper we tackle the problem of wind speed estimation in a given point (wind farm), from the corresponding synoptic pressure pattern. The problem involves daily pressure patterns in a synoptic grid, in this case centred in Spain, and a wind speed module measure. The first novelty of the paper is that this wind speed is discretized into different levels of wind (classes) in order to treat it as a classification problem. The motivation behind this is that the manager of the wind farm can get enough information from the considered classes in order to set functional operations for the farm (such as wind turbines stop, for example). Note that the exact wind speed value is not usually important for this task. Four classes have been considered because they cover all the wind speed spectrum measured in a given wind farm.

Ordinal classification plays an important role in various decision making tasks. In these tasks, the classes are ordered. However, little attention is paid to this type of learning tasks compared with traditional nominal classification learning (where no order is found between the classes). The characteristics of the wind make that the problem can be defined as an ordinal classification problem, in which the different classes (wind speed intervals), can be ordered from the smallest to the largest, in increasing order. The second novel point of this paper is the use of the ordering information for obtaining better quality classifiers, and the comparison of their performance with respect to nominal classifiers.

We have tested different intelligent algorithms to this task, and we compare the results of the different approaches for three wind farms in Spain, obtaining interesting results and conclusions.

The structure of the rest of the paper is the following: next section presents the definition of the problem. Section III presents the main characteristics of the algorithms tested for this problem. The experiments of the paper are then presented

in Section IV. Finally, Section V closes the paper giving some concluding remarks.

II. PROBLEM DEFINITION

The problem in this paper may be summarized as follows: Let $\mathbf{y} = \{y_i, i = 1, \dots, T\}$ be a series of daily wind speed discretized measures at a given point, in such a way that $y_i \in Y = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_K\}$, i.e. y belongs to one out of K classes which are subjected to an ordinal order (i.e. $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \dots \prec \mathcal{C}_K$, where \prec is an ordering relationship between the labels). Let $\mathbf{X} = \{\mathbf{x}_i, i = 1, \dots, T\}$ be a series of daily synoptic-scale pressure measures in a grid. In our case, each component of \mathbf{X} is a matrix of 14×13 surface pressure values (182 values), measured in a grid surrounding the Iberian Peninsula (Figure 1). The problem we face in this paper is a classification problem, consisting of obtaining a machine Φ by using a training set $\{(\mathbf{x}_i, y_i), i = 1, \dots, T_t < T\}$ (the first part of the series), so that for a given value of \mathbf{x}_i , it estimates the associated value of y_i , i.e. $\Phi(\mathbf{x}_i) \rightarrow y_i$, in such a way that the machine Φ minimizes an error measure in an independent test set $\{(\mathbf{x}_i, y_i), i = T_t + 1, \dots, T\}$ (the rest of the series), to ensure the good generalization of the machine.

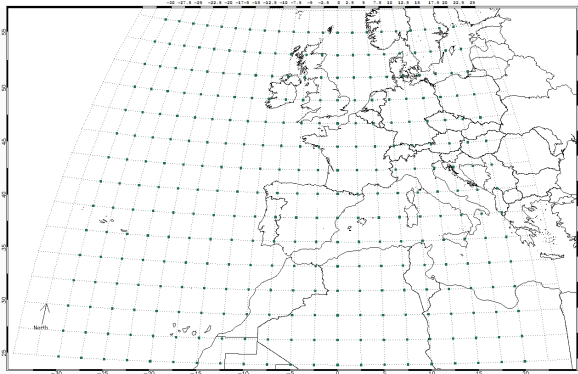


Fig. 1. Synoptic pressure grid considered (Sea Level Pressure values have been used in this paper).

Two evaluation metrics have been considered which quantify the accuracy of n predicted ordinal labels for a given dataset $\{y_1^*, y_2^*, \dots, y_n^*\}$, with respect to the true targets $\{y_1, y_2, \dots, y_n\}$:

- 1) Accuracy (C) is simply the fraction of correct predictions on individual samples:

$$C = \frac{1}{n} \sum_{i=1}^n I(y_i^* = y_i), \quad (1)$$

where $I(\cdot)$ is the zero-one loss function and n is the number of patterns of the dataset.

- 2) Mean Absolute Error (MAE) is the average deviation of the prediction from the true target, i.e.:

$$MAE = \frac{1}{n} \sum_{i=1}^n |\mathcal{O}(y_i^*) - \mathcal{O}(y_i)|, \quad (2)$$

where $\mathcal{O}(\mathcal{C}_k) = k, 1 \leq k \leq K$, i.e. $\mathcal{O}(y_i)$ is the order of class label y_i .

These measures are aimed to evaluate two different aspects that can be taken into account when an ordinal regression problem is considered: whether the patterns are generally well classified (C) and whether the classifier tends to predict a class as close to the real class as possible (MAE).

III. EVALUATED CLASSIFIERS

In this paper, several methods have been tested for facing the problem described in Section II. As mentioned in Section I, one of the aims of this paper is to evaluate the improvement of a standard classifier when including the label ordering information in its description. In this way, the evaluated classifiers have been organized in two different groups, nominal classifiers and ordinal classifiers. Support Vector Machine (SVM) methods receive a special attention because they yield the best performance for the problem of wind speed prediction (see Section IV).

A. Nominal classifiers

Very well-known standard nominal classifiers have been considered. We briefly describe their main characteristics in the following subsections.

1) *Support Vector Machines*: The SVM [2], [5] is perhaps the most common kernel learning method for statistical pattern recognition. They are linear parametric models re-cast into an equivalent dual representation in which the predictions are based on a linear combination of a kernel function evaluated at the training data points. The parameters of the kernel model are typically given by the solution of a convex optimization problem, so there is a single, global optimum.

SVM [2] can be thought as generalized perceptrons with a kernel that computes the inner product on transformed input vectors $\phi(\mathbf{x})$, where $\phi(\mathbf{x})$ denote the feature vector \mathbf{x} in a high dimensional reproducing kernel Hilbert space (RKHS) related to \mathbf{x} by a specific transformation. All computations are done using the reproducing kernel function only, which is defined as:

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}) \cdot \phi(\mathbf{x}') \rangle, \quad (3)$$

where $\langle \cdot \rangle$ denotes inner product in the RKHS.

The basic idea behind SVMs is to separate the two different classes — they are firstly defined for two classes and then extended to the multiclass case — through a hyperplane which is specified by its normal vector \mathbf{w} and the bias b . The hyperplane can be given as:

$$\langle \mathbf{w} \cdot \phi(\mathbf{x}) \rangle + b = 0, \quad (4)$$

what yields the corresponding decision function:

$$f(\mathbf{x}) = y^* = \text{sgn}(\langle \mathbf{w} \cdot \phi(\mathbf{x}) \rangle + b), \quad (5)$$

where $y^* = +1$ if \mathbf{x} belongs to the corresponding class and $y^* = -1$ otherwise.

Beyond specifying non-linear discriminants by kernels, another generalization has been proposed which replaces hard

margins by soft margins. This way allows to handle noise and pre-labeling errors, which often occur in practice. *Slack-variables* ξ_i are used to relax the hard-margin constraint [5].

As Vapnik [5] shows, the optimal separating hyperplane is the one which maximizes the distance between the hyperplane and the nearest points of both classes (called margin) and results in the best prediction for unseen data. In this way, the optimal separating hyperplane with maximal margin can be formulated as the following quadratic optimization problem:

$$\min_{\mathbf{w} \in \mathbb{R}^n, \xi \in \mathbb{R}^n} L(\mathbf{w}, \xi) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i, \quad (6)$$

subject to:

$$y_i \cdot (\langle \mathbf{w} \cdot \phi(\mathbf{x}) \rangle + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i = 1, \dots, n, \quad (7)$$

where y_i is the class of the input pattern \mathbf{x}_i .

In order to deal with the multiclass case, a “1-versus-1” approach can be considered, following the recommendations of Hsu and Lin [13]. The idea is to construct a binary classifier per each of pair of classes and joining their multiple responses to obtain a final prediction.

2) *Other standard nominal classifiers*: Other standard machine learning classifiers have been considered, given their good performance and competitiveness. They include:

- The Logistic Model Tree (LMT) [18] classifier.
- The C4.5 classification tree inducer [25].
- The AdaBoost.M1 algorithm [9], using C4.5 as the base learner and the maximum number of iterations set to 10 and 100 iterations (Ada10 and Ada100).
- Multi-logistic regression methods, including the MultiLogistic (MLogistic) and SimpleLogistic (SLogistic) algorithms.
 - MLogistic is an algorithm for building a multinomial logistic regression model with a ridge estimator to guard against over-fitting by penalizing large coefficients, based on the work by le Cessie and van Houwelingen [4]. In order to find the coefficient matrices, a Quasi-Newton Method is used. Specifically, the method used is the active-sets’ method with the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update.
 - SLogistic algorithm builds multinomial logistic regression models by using the LogitBoost algorithm [11], which was proposed by Friedman et al. for fitting *additive logistic regression models* by maximum likelihood. These models are a generalization of the (linear) logistic regression models. This version of the algorithm is based on controlling the number of variables of the model to avoid over-fitting [18].

B. Ordinal Classifiers

In an ordinal regression problem, an example (\mathbf{x}, y) is composed of an input vector $\mathbf{x} \in \mathbb{R}^n$ and an ordinal label (i.e., rank) $y_i \in Y = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_K\}$, in such a way that $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \dots \prec \mathcal{C}_K$. This looks similar to that of a multiclass classification problem, except that the ranks are ordered.

1) *A Simple Approach to ordinal regression (ASA)*: It is straightforward to realize that ordinal information allows ranks to be compared. For a fixed rank $\mathcal{O}(y_k) = k$, an associated question could be “is the rank of \mathbf{x} greater than k ?”. Such a question is exactly a binary classification problem, and the rank of \mathbf{x} can be determined by asking multiple questions for $k = 1, 2, \dots, (K - 1)$. Frank and Hall [8] proposed to solve each binary classification problem independently and combine the binary outputs to a rank.

2) *Extended Binary Classification (EBC)*: Although the approach proposed by Frank and Hall [8] is simple, the generalization performance using the combination step cannot be easily analyzed. The EBC method [19] works differently. First, all the binary classification problems are solved jointly to obtain a single binary classifier. Second, a simpler step is used to convert the binary outputs to a rank, and generalization analysis can immediately follow.

Let us assume that $f(\mathbf{x}, k)$ is a binary classifier for all the associated questions above. A good prediction would be the following: $f(\mathbf{x}, k) = 1$ (“yes”) for $k = 1$ to $k = y - 1$ (where y is the rank associated to the pattern \mathbf{x}) and $f(\mathbf{x}, k) = 0$ (“no”) afterwards.

Furthermore, the ordinal information can help to model the relative confidence in the binary outputs. That could be possible if we associate the absolute value of $f(\mathbf{x}, k)$ to the confidence of the outputs. A possible ranking function $r(\mathbf{x})$ based on all the binary answers $f(\mathbf{x}, k)$ is the following:

$$r(\mathbf{x}) = 1 + \sum_{k=1}^{K-1} \llbracket f(\mathbf{x}, k) > 0 \rrbracket, \quad (8)$$

being $\llbracket \cdot \rrbracket$ a Boolean test which is 1 if the inner condition is true, and 0 otherwise. In summary, the EBC method is based on the following three steps:

- 1) Transform all training samples (\mathbf{x}_i, y_i) into extended samples $(\mathbf{x}_i^{(k)}, y_i^{(k)})$, $1 \leq k \leq K - 1$:

$$\mathbf{x}_i^{(k)} = (\mathbf{x}_i, k), \quad y_i^{(k)} = 2\llbracket k < \mathcal{O}(y_i) \rrbracket - 1, \quad (9)$$

but weighting these samples in the following way:

$$w_{y_i, k} = |C_{\mathcal{O}(y_i), k} - C_{\mathcal{O}(y_i), k+1}|, \quad (10)$$

where C is a V-shaped cost matrix, with $C_{\mathcal{O}(y_i), k-1} \geq C_{\mathcal{O}(y_i), k}$ if $k \leq \mathcal{O}(y_i)$ and $C_{\mathcal{O}(y_i), k} \leq C_{\mathcal{O}(y_i), k+1}$ if $k \geq \mathcal{O}(y_i)$.

- 2) All the extended examples are then jointly learned by a binary classifier f with confidence outputs, aiming at a low weighted 0/1 loss.
- 3) The ranking rule (8) is used to construct a final prediction for new samples.

This framework can be adapted for Support Vector Machines, by using a threshold model to estimate $f(\mathbf{x}, k)$:

$$f(\mathbf{x}, k) = g(\mathbf{x}) - \theta_k, \quad (11)$$

where $g(\mathbf{x})$ is a non-linear function defined as $g(\mathbf{x}) = \langle \mathbf{w} \cdot \phi(\mathbf{x}) \rangle$.

As long as the threshold vector θ is ordered, i.e., $\theta_1 < \theta_2 < \dots < \theta_{K-1}$, the function f is rank-monotonic. The adaptation of the SVM framework can be performed by simply defining extended kernels. The extended kernels of the extended examples (\mathbf{x}, k) will be the original kernel plus the inner product between the extensions:

$$K((\mathbf{x}, k), (\mathbf{x}', k')) = \langle \phi(\mathbf{x}) \cdot \phi(\mathbf{x}') \rangle + \langle \mathbf{e}_k \cdot \mathbf{e}_{k'} \rangle, \quad (12)$$

where \mathbf{E} is a coding matrix of $(K - 1)$ rows and \mathbf{e}_k is the k -th row of this matrix. Depending on the selection of \mathbf{E} , several SVM algorithms can be reproduced. In this paper, we use $\mathbf{E} = \mathbf{I}_{K-1}$ and the absolute value cost matrix, applied to the standard soft-margin SVM, so:

$$f(\mathbf{x}^{(k)}) = \langle (\mathbf{u}, \theta), (\phi(\mathbf{x}), \mathbf{e}_k) \rangle. \quad (13)$$

IV. EXPERIMENTS

In the following subsections, the description of the datasets and the experimental design is given. Then, the details on the preprocessing of the datasets are explained, and finally the results are included.

A. Dataset Description and Experimental Design

Three different wind farms have been considered for this study, resulting in three datasets (M, U and Z). Each dataset includes a series of discretized wind speed values (targets), taken in a tower at 40m of height, and averaged over 24 hours to obtain daily data values. On the other hand, a series of grids of average daily pressure maps for the same period have been obtained from the National Center for Environmental Prediction/National Center for Atmospheric Research Reanalysis Project (NCEP/NCAR) [16], [23], which are public data profusely used in climatology and meteorology applications. As previously mentioned, we have considered an uniform grid in latitude and longitude, shown in Figure 1, with 182 measurement points, and each element of this grid is one input variable.

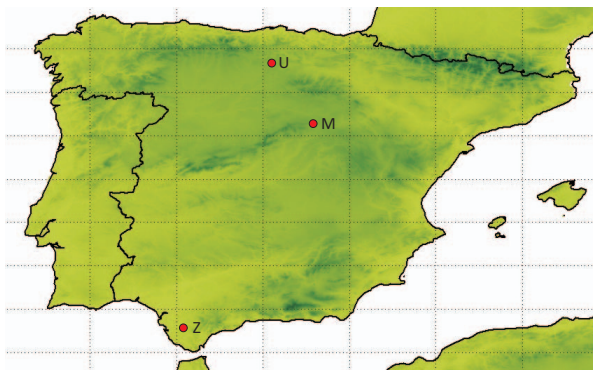


Fig. 2. Location of the wind farms considered in this work.

For each wind farm, two different sets are obtained, one for training the models and another one for assessing the performance of the algorithms. In this way, the structure of the different datasets used in this study is given in Table I.

TABLE I
STRUCTURE OF TRAINING AND TEST SETS: TOTAL NUMBER OF PATTERNS (Size), NUMBER OF PATTERN IN EACH CLASS (C_1, C_2, C_3, C_4) (Distribution) AND FINAL NUMBER OF PRINCIPAL COMPONENTS (PCs)

Wind farm	Training		Test		PCs ¹
	Size	Distribution	Size	Distribution	
M	2231	(220,1590,396,52)	1115	(173,779,147,16)	10
U	2017	(527,1167,280,43)	1008	(361,547,85,15)	6
Z	1749	(901,637,184,27)	874	(516,279,68,11)	13

The structures of these datasets are challenging, because the distribution of the different classes is clearly imbalanced, with very few situations of high wind speed (class C_4) and lot of patterns belonging to a moderate wind speed class (class C_2).

Since all the tested algorithms are deterministic, they will be run once, deriving a model from the training set and evaluating its accuracy over the test set. Both training and test sets are parts of a wind series, so it is not advisable to do different random partitions of them.

For the selection of the SVM's hyperparameters (regularization parameter, C , and width of the Gaussian functions, γ), a grid search algorithm was applied with a ten-fold cross-validation, using the following ranges: $C \in \{10^{-3}, 10^{-2}, \dots, 10^3\}$ and $\gamma \in \{10^{-3}, 10^{-2}, \dots, 10^3\}$. This cross-validation has been applied only taking into account the training data, and then repeating the process with the best parameter combination using the complete training set.

B. Preprocessing of the dataset

As previously said, our vector of inputs is formed by 14×13 surface pressure values (182) values in a grid around the Iberian Peninsula, which results in a very high number of variables. When too many inputs are presented to the standard machine learning algorithms, a very well known problem appears, the *curse of dimensionality*, which can decrease the performance of these algorithms and significantly increase the computational cost.

With the aim of alleviating this problem, a simple approach has been applied based on the standard technique of Principal Component Analysis (PCA) [15]. PCA is the predominant linear dimensionality reduction technique, and has been widely applied to datasets in all scientific domains. Generally speaking, PCA maps data points from a high dimensional space to a low dimensional space, while keeping all the relevant linear structure intact.

PCA algorithm returns so many principal components (PCs, linear combinations of the input variables) as the total number of inputs, but they are sorted in the following way: the first PC has as high variance as possible (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it will be orthogonal to (i.e. uncorrelated with) the preceding components. One should still decide how many of the first PCs are retained when reducing the dimensionality of the problem.

With this aim, we have applied the algorithm included in

Deciding # of Principal Components:

Input: Training dataset (Tr), Test dataset (Te)

Output: Projected training dataset (Tr^*), Projected test dataset (Te^*)

- 1: Apply PCA to Tr , without considering Te
- 2: $Max \leftarrow$ Number of PCs retaining a 99% of the total variance of the dataset
- 3: **for** $i = 1 \rightarrow Max$ **do**
- 4: $Tr_i \leftarrow Tr$ projected over the i first PCs.
- 5: Apply a ten-fold cross-validation method, considering Tr_i data and the LDA classifier.
- 6: $e_i \leftarrow$ cross-validated error of the classifier.
- 7: **end for**
- 8: $n \leftarrow \operatorname{argmin}_i e_i$
- 9: $Tr^* \leftarrow Tr$ projected over the n first PCs.
- 10: $Te^* \leftarrow Te$ projected over the n first PCs.
- 11: **return** Tr^* and Te^*

Fig. 3. Algorithm for deciding the number of principal components

Fig. 3. The idea is very simple, the coefficients of the PCs are obtained using the training data and we try all possible combinations from 1 to the number of PCs that retain a 99% of the variance. A 10-fold cross-validation is applied for each combination, estimating the error with one of the simplest classifier possible in order to limit the computational time (a Linear Discriminant Analysis, LDA). Once the best number of PCs is decided, training and test data are projected into them, and the reduced datasets are returned.

C. Results

The results for the two different evaluation measures considered (C and MAE , see Equations (1) and (2)) are included in Tables II and III, respectively. Based on the C and MAE values, the ranking of each method in each park is obtained ($R = 1$ for the best performing method and $R = 9$ for the worst one). The mean accuracy and MAE (\bar{C} and \bar{M}) as well as the mean ranking (\bar{R}_C and \bar{R}_M) are also included in Tables II and III. The first conclusion is that very high accuracies are obtained, what reveals that considering the problem as a classification task can provide an accurate information of the wind farm (with a lot of values higher than 70% of well predicted samples).

From these tables, the SVM methods seems to be the most competitive ones from all the different alternatives considered. If we analyse the mean ranking and performance the EBC(SVM) methodology obtains the better results for both measures, the second best method being standard SVM. However, high accuracy values can be masking a lower ranking performance (i.e. a high MAE value), because the classifier can tend to assign rank values far from the actual ones.

To determine the statistical significance of the rank differences observed for each method in the different datasets, a non-parametric Friedman test [10] has been carried out with the C and MAE rankings of the different methods (since a previous evaluation of the C and MAE values results in

TABLE II
TEST ACCURACY ($C(\%)$) RESULTS OBTAINED BY USING THE DIFFERENT METHODS EVALUATED

Type of classifier	Classifier	Wind farm				
		M	U	Z	$\bar{C}(\%)$	\bar{R}_C
Nominal	SVM	73.45	62.50	69.45	68.47	2.67
	C4.5	70.04	57.54	55.72	61.10	7.67
	Ada10(C4.5)	68.43	63.39	61.10	64.31	5.67
	Ada100(C4.5)	72.20	62.50	66.36	67.02	3.33
	LMT	70.94	62.80	64.65	66.13	4.00
	MLogistic	71.66	57.44	62.24	63.78	5.67
Ordinal	SLogistic	71.21	57.34	62.36	63.64	6.17
	ASA(C4.5)	70.76	57.34	56.75	61.62	7.83
	EBC(SVM)	73.90	62.50	70.48	68.96	2.00

The best result is in bold face and the second best result in italics

TABLE III
TEST MEAN ABSOLUTE ERROR (MAE) RESULTS OBTAINED BY USING THE DIFFERENT METHODS EVALUATED

Type of classifier	Classifier	Wind farm				
		M	U	Z	\bar{M}	\bar{R}_M
Nominal	SVM	0.265	0.381	0.314	0.320	2.17
	C4.5	0.310	0.434	0.487	0.410	8.17
	Ada10(C4.5)	0.318	0.382	0.420	0.373	5.67
	Ada100(C4.5)	0.281	0.389	0.354	0.341	3.67
	LMT	0.293	0.383	0.373	0.350	4.50
	MLogistic	0.288	0.433	0.405	0.375	5.33
Ordinal	SLogistic	0.293	0.434	0.400	0.376	6.00
	ASA(C4.5)	0.299	0.438	0.463	0.400	8.00
	EBC(SVM)	0.261	0.382	0.295	0.313	1.50

The best result is in bold face and the second best result in italics

rejecting the normality and the equality of variances hypothesis). The test shows that the effect of the method used for classification is statistically significant at a significance level of $\alpha = 5\%$, as the confidence interval is $C_0 = (0, F_{0.05} = 2.09)$ and the F-distribution statistical values are $F^* = 2.92 \notin C_0$ for C and $F^* = 5.01 \notin C_0$ for MAE . Consequently, the null-hypothesis stating that all algorithms perform equally in mean ranking is rejected.

It has been noted that the approach that compares all classifiers to each other in a post-hoc test is not as sensitive as the approach comparing all classifiers to a given classifier (a control method). The Bonferroni-Dunn test [7] is an example of this latter type of comparison with a control method. This test has been applied to both C and MAE rankings using EBC(SVM) as the control method. The test concludes that there are not significant differences when comparing C values, and that the differences are significant at a significance level of $\alpha = 5\%$ when EBC(SVM) is compared to C4.5 and ASA(C4.5) (with differences of ranking of 6.67 and 6.50, respectively).

Consequently, an important conclusion of our study is that the use of the ordering information improves the results obtained by the nominal classifiers, specially when taking into account the MAE measure: EBC(SVM) improves the accuracy and MAE values of standard SVM for the three wind farms considered; ASA(C4.5) also improves the accuracy and MAE values of standard C4.5 method for all the wind farms except for the U wind farm.

V. CONCLUSIONS

This paper introduced a new approach for predicting wind speed, based on a classification task rather than the usual regression approach. Wind speed was discretized in four different ranges, which gather the main information needed by the experts when managing the wind farm. On the other hand, synoptic pressure measures have been considered as the input variables. The results of this preliminary study show that the best performing method is the SVM, with very high accuracy and low *MAE* values. This paper has also shown how ordering information (more precisely, the EBC algorithm) can still improve the performance of SVM, yielding to more accurate predictions.

ACKNOWLEDGEMENT

This work has been partially supported by Spanish Ministry of Industry, Tourism and Trading, under an Avanza 2 project, number TSI-020100-2010-663, the TIN 2008-06681-C06-03 project of the Spanish Inter-Ministerial Commission of Science and Technology (MICYT), FEDER funds and the P08-TIC-3745 project of the "Junta de Andalucía" (Spain).

REFERENCES

- [1] T.G. Barbounis and J.B. Theocharis, "Locally recurrent neural networks for long-term wind speed and power prediction," *Neurocomputing*, vol. 69, no. 4-6, pp. 466-496, 2006.
- [2] B. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, D. Haussler, Ed. Pittsburgh, PA: ACM Press, 1992, pp. 144-152.
- [3] M. Burlando, M. Antonelli and C. F. Ratto, "Mesoscale wind climate analysis: identification of anemological regions and wind regimes," *International Journal of Climatology*, vol. 28, pp. 629-641, 2008.
- [4] S. le Cessie and J. van Houwelingen, "Ridge estimators in logistic regression," *Applied Statistics*, vol. 41, no. 1, pp. 191-201, 1992.
- [5] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.
- [6] Z.H. Chen, S.Y. Cheng, J.B. Li, X.R. Guo, W.H. Wang and D.S. Chen, "Relationship between atmospheric pollution processes and synoptic pressure patterns in northern China," *Atmospheric Environment*, vol. 42, no. 24, pp. 6078-6087, 2008.
- [7] J. Demar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1-30, 2006.
- [8] E. Frank and M. Hall, "A simple approach to ordinal classification," in *EMCL 01: Proceedings of the 12th European Conference on Machine Learning*. London, UK: Springer-Verlag, 2001, pp. 145-156.
- [9] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the Thirteenth International Conference on Machine Learning*. Morgan Kaufmann, 1996, pp. 148-156.
- [10] M. Friedman, "A comparison of alternative tests of significance for the problem of *m* rankings," *Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86-92, 1940.
- [11] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *The Annals of Statistics*, vol. 38, no. 2, pp. 337-374, 2000.
- [12] Gomez-Muñoz, V. M. and M. A. Porta-Gándara, "Local wind patterns for modeling renewable energy systems by means of cluster analysis techniques," *Renewable Energy*, vol. 2, pp. 171-182, 2002.
- [13] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multi-class support vector machines," *IEEE Transaction on Neural Networks*, vol. 13, no. 2, pp. 415-425, 2002.
- [14] P. A. Jiménez, J. F. Gonzalez-Rouco, J. P. Montavez, E. García-Bustamante and J. Navarro, "Climatology of wind patterns in the northeast of the Iberian Peninsula," *International Journal of Climatology*, vol. 29, pp. 501-525, 2009.
- [15] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer, Oct. 2002
- [16] E. Kalnay et al. "The NCEP/NCAR reanalysis project," *Bulletin of the American Meteorological Society*, vol. 77, pp. 437-471, 1996.
- [17] M. Khashei, M. Bijari and G. Raissi-Ardali, "Improvement of Auto-Regressive Integrated Moving Average Models Using Fuzzy Logic and Artificial Neural Networks (ANNs)," *Neurocomputing*, vol. 72, no. 4-6, pp. 956-967, 2009.
- [18] N. Landwehr, M. Hall, and E. Frank, Logistic model trees, *Machine Learning*, vol. 59, no. 1, pp. 161-205, 2005.
- [19] L. Li and H.-T. Lin, "Ordinal regression by extended binary classification," in *Advances in Neural Information Processing Systems 19*, 2007, pp. 865-872.
- [20] G. Li and J. Shi, "Application of Bayesian model averaging in modeling long-term wind speed distributions," *Renewable Energy*, vol. 35, no. 6, pp. 1192-1202, 2010.
- [21] A. Mellit, S. A. Kalogirou, L. Hontoria and S. Shaari, "Artificial intelligence techniques for photovoltaic applications: A review," *Progress in Energy and Combustion Science*, vol. 34, no. 5, pp. 406-419, 2009.
- [22] M. A. Mohandes, T. O. Halawani, S. Rehman and A. A. Hussain, "Support vector machines for wind speed prediction," *Renewable Energy*, vol. 29, pp. 939-947, 2004.
- [23] <http://www.esrl.noaa.gov/psd/data/reanalysis/reanalysis.shtml>
- [24] D. Paredes, D. R. M. Trigo, R. García-Herrera and I. F. Trigo, "Understanding precipitation changes in Iberia in early spring: weather typing and storm tracking approaches," *Journal of Hydrometeorology*, vol. 7, pp. 101-113, 2006.
- [25] J. R. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.
- [26] R. Romero, G. Summer, C. Ramis and A. Genoves, "A classification of the atmospheric circulation patterns producing significant daily rainfall in the Spanish Mediterranean area," *International Journal of Climatology*, vol. 19, pp. 765-785, 1999.
- [27] J.L. Torres, A. García, M. De Blas and A. De Francisco, "Forecast of hourly average wind speed with ARMA models in Navarre (Spain)," *Solar Energy*, vol. 79, pp. 65-77, 2005.
- [28] R. M. Trigo and C. C. DaCamara, "Circulation weather types and their influence on the precipitation regime in Portugal," *International Journal of Climatology*, vol. 20, pp. 1559-1581, 2000.