

ORIGINAL RESEARCH

Open access

Multi-Horizon Wind Speed Prediction Using Ordinal Neural Networks and Soft-Label Optimization

Arjun Nair¹, Kavita Rao^{1*}, Sameer Patel², Rohan Bhat²

Abstract

This paper introduces and assesses two innovative ordinal classification approaches for forecasting wind speed across three horizons: 1h, 4h, and 8h. To tackle the issue, wind speed values are categorized into four discrete classes, which are vital for effective wind farm operations. Each class conveys key information about wind farm output, ranging from very low wind speeds to extreme wind speeds and their associated generation scenarios, thereby supporting operational decisions by wind farm managers. Ordinal classifiers prove more appropriate than nominal ones for this task. The central goal of the study is to evaluate recently developed ordinal classifiers for wind speed forecasting, with particular emphasis on extreme wind conditions that frequently lead to turbine shutdowns. The analysis relies on hourly wind speed records from a wind farm in Spain, along with predictor variables derived from the European Centre for Medium-Range Weather Forecasts Reanalysis v5 (ERA5). The proposed techniques consist of an Artificial Neural Network (ANN) employing the Cumulative Link Model as its ordinal output mechanism (MLP-CLMO), which prioritizes overall effectiveness, and an ANN refined through a soft labeling strategy grounded in triangular distributions (MLP-TO), which performs particularly well for extreme class predictions. The findings show that both methods outperform other nominal and ordinal techniques across evaluation metrics that properly address the data's imbalance and ordinal characteristics. MLP-CLMO outperforms MLP-TO across general and ordinal performance, whereas MLP-TO shows stronger performance in predicting the extreme class.

Keywords Extreme wind speed prediction, Wind energy, Ordinal classification problems, Cumulative link models, Soft-labeling, Unimodal distributions

*Correspondence:

Kavita Rao

kavita.rao@iitm.ac.in

¹ Department of Smart Energy Systems, Faculty of Engineering, IIT Madras, Chennai, India

² Department of Grid Analytics and Optimization, Faculty of Technology, IIT Bombay, Mumbai, India

Introduction

Over the past five years, worldwide energy demand has surged to unprecedented levels, driven primarily by rapid industrial expansion and urbanization globally, aimed at economic recovery following the COVID-19 pandemic [1], particularly in major economies such as China [2]. This rise in energy needs occurs alongside efforts in developed nations to decarbonize their economies in response to escalating global warming caused by human-induced climate change over the last 50 years [3]. Consequently, traditional fossil fuel-based energy sources, which carry substantial environmental consequences, cannot meet this growing global demand sustainably [4]. Renewable energy (RE) sources must therefore assume a leading role in the coming decades to establish a more dependable, cleaner, and sustainable energy mix.

In particular, wind energy offers an environmentally friendly energy option, supported by well-established technology deployed worldwide, delivering zero greenhouse gas emissions, minimal ecological disruption, and comparatively low

costs [5]. At present, wind energy ranks among the leading contributors to economic decarbonization in developed countries. Projections indicate that by 2030, wind power will account for 22% of global electricity supply [6], with even greater expectations projected for 2050 [7]. The bulk of wind power generation currently occurs at wind farm installations, with the electricity transmitted to urban areas for integration into existing power grids [8]. The primary challenge with wind energy, as with other renewable sources, lies in its reliance on weather patterns and its variable, intermittent nature, which makes it difficult to deliver consistent power. Extensive research efforts are underway to achieve greater integration of wind energy into electrical systems, concentrating mainly on enhancing wind farm efficiency in power output [9], safeguarding power systems against cyberattacks [10], advancing the incorporation of battery energy storage in wind farms [11], and refining wind power forecasting systems [12].

Wind speed and associated power output forecasting at wind farms, therefore, remains a prominent research area that has drawn considerable interest from scientists over recent decades [13]. In numerous studies, wind speed or wind power forecasting in wind farms has been approached as a regression problem [14], employing numerical models [15] or Artificial Intelligence techniques [16], across various prediction time horizons [17]. In recent years, advanced Deep Learning (DL) techniques [18] and data-driven strategies [19] have also been applied to wind speed forecasting. Nevertheless, wind speed prediction has likewise been framed as a classification problem, particularly within wind farm settings, where multi-class classification addresses wind speed and power generation [20], and binary classification targets extreme wind speed events [21]. Additional classification approaches appear in the wind speed prediction literature, including [22], which compares ordinal and nominal classifiers for wind speed forecasting, [23], which applies both nominal and ordinal classifiers to wind speed prediction, [24], which utilizes ordinal classification for wind speed categorization based on synoptic pressure patterns as inputs, and [25], which employs ordinal classification for predicting wind ramp events at wind farms.

In this paper, we assess the effectiveness of innovative ordinal classifiers applied to wind speed forecasting in wind farms, with direct relevance to power generation. In particular, we introduce two new ordinal classification methods that combine Artificial Neural Networks (ANNs) with approaches tailored for ordinal targets: (1) MLP-CLMO, an ANN that incorporates the Cumulative Link Model (CLM) [26], mapping each input pattern to a one-dimensional latent variable and assigning the ordinal class through learned thresholds applied to this projection; and (2) MLP-TO, an ANN trained using soft labels generated from triangular distributions [27], which substitute conventional one-hot labels with probability distributions that preserve the ordinal relationships among classes.

The key distinction of MLP-CLMO from other ordinal neural network approaches is its direct integration of the CLM into the ANN, resulting in a threshold-based output layer that explicitly models the ordinal nature of the target variable—unlike standard classifiers, which rely on a softmax layer. Compared with a conventional ANN classifier using softmax and categorical cross-entropy loss, MLP-CLMO relies on a latent projection, a series of thresholds, a link function, and explicit modeling of cumulative probabilities over ordered classes. By contrast, MLP-TO adjusts the loss function via unimodal regularization based on triangular distributions, thereby capturing ordinal class relationships without altering the underlying network structure.

Both proposed methods offer substantial benefits over conventional ordinal classification techniques. MLP-CLMO offers great flexibility, as it can be incorporated into virtually any ANN architecture simply by replacing the softmax output layer with a CLM-based layer, requiring only minor structural modifications. MLP-TO is even more versatile, since it solely modifies the loss function via unimodal regularization and thus requires no architectural adjustments. Moreover, it introduces just one extra hyperparameter, which simplifies its implementation and tuning. These features set MLP-CLMO and MLP-TO apart from other ordinal techniques, such as LROAT and LROIT [28], which are generally more constrained in scope and more elaborate in their construction. In particular, MLP-CLMO prioritizes overall performance, while MLP-TO emphasizes accurate prediction of the extreme classes.

We evaluated these novel methods on a wind speed prediction task formulated as multi-class ordinal classification, with four classes defined to represent the most relevant wind regimes and corresponding power production scenarios in a

wind farm: very low wind speeds linked to negligible or zero production (below 6 m/s), winds yielding low production (6–12 m/s), winds producing high output (12–20 m/s), and extreme winds that deliver high production but carry the risk of turbine shutdown (above 20 m/s). The proposed ordinal classifiers achieve outstanding overall results across the three forecasting horizons considered (1 h, 4 h, and 8 h), as measured by three performance metrics that properly account for both the ordinal nature and the class imbalance of the data. Additionally, both approaches—and especially MLP-TO—show strong accuracy in predicting the minority class associated with extreme wind speeds, with their advantages becoming more evident at longer prediction horizons.

Materials and Methods

This section first provides a formal introduction to ordinal classification and then outlines the CLM and soft-labeling techniques applied in this study to the wind-speed prediction problem.

Ordinal classification

Within the framework of Machine Learning (ML), a classification problem with J classes involves assigning the appropriate class label (hard label) $y_i \in Y = \{C_1, C_2, \dots, C_J\}$ to each instance characterized by an input feature vector $x_i \in X \subseteq \mathbb{R}^d$, where $d \in \mathbb{N}$ denotes the dimensionality of the feature space. Accordingly, a classification dataset can be represented as $D = (X, y) = \{(x_i, y_i); i = 1, 2, \dots, N\}$, where N is the total number of instances, $X = \{x_1, x_2, \dots, x_N\}$ is the matrix of all input vectors, and $y = \{y_1, y_2, \dots, y_N\}$ is the vector of corresponding target labels. Unlike nominal classification, ordinal classification (also known as ordinal regression) [29] assumes a natural ordering among the classes. This applies to the wind speed prediction problem addressed here, where wind speeds are discretized into four ordered classes: below 6 m/s, 6–12 m/s, 12–20 m/s, and above 20 m/s. Employing ordinal methods can enhance predictive accuracy by leveraging this ordering information, whereas nominal approaches ignore it [30].

The ordering relation among classes can be denoted as $C_1 < C_2 < \dots < C_J$, where $< C_j$ indicates the precedence. The position of class j is then given by the integer $O(C_j) = j$, satisfying $O(C_1) < O(C_2) < \dots < O(C_J)$. However, unlike regression tasks, the precise distances between classes remain unknown in ordinal classification. Moreover, misclassifications between non-adjacent classes should incur greater penalties than those between adjacent classes, with the penalty increasing in proportion to the separation between classes.

Cumulative link model (CLM)

CLMs [26] rely on the notion of a latent variable projected onto a one-dimensional space, combined with a series of thresholds that partition this projection into ordinal categories. We adopt the CLM due to its proven strong performance in ordinal classification tasks [26, 31] and its relatively straightforward implementation. To apply this technique, we utilize a MultiLayer Perceptron (MLP)-based ANN equipped with a CLM-based ordinal output. The structure of the proposed model, MLP-CLMO, is depicted in Figure 1.

As shown, the input layer of MLP-CLMO consists of d neurons (corresponding to each feature in x_i), while the number of hidden neurons m is optimized during training. The output layer comprises a single neuron whose value feeds into the CLM for ordinal prediction. Formally, the output neuron for the i th instance in MLP-CLMO is expressed as:

$$f(x_i, \Theta) = f(x_i, W, \beta) = \beta_0 + \sum_l m_l = 1 \beta_l Bl(x_i, w_l), \quad (1)$$

Where $\Theta = (W, \beta)$ denotes the set of trainable parameters, $W = \{w_1, w_2, \dots, w_m\}$ contains the weights connecting the input layer to the m hidden neurons, and $\beta = \{\beta_0, \beta_1, \dots, \beta_m\}$ includes the weights from the hidden layer to the output (with β_0 as the bias term). The term $Bl(x_i, w_l)$ represents the sigmoid activation function for the l -th hidden neuron, which introduces non-linearity to the input vector $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$ and is defined as:

$$Bl(xi, wl) = 11 + e - (wl0 + \sum dk = 1wlkxik), l = 1, \dots, m, 0 < Bl(xi, wl) < 1, (2)$$

Where $wTl = \{wl0, wl1, wl2, \dots, wld\}$ incorporates $wl0$ as the bias for that neuron.

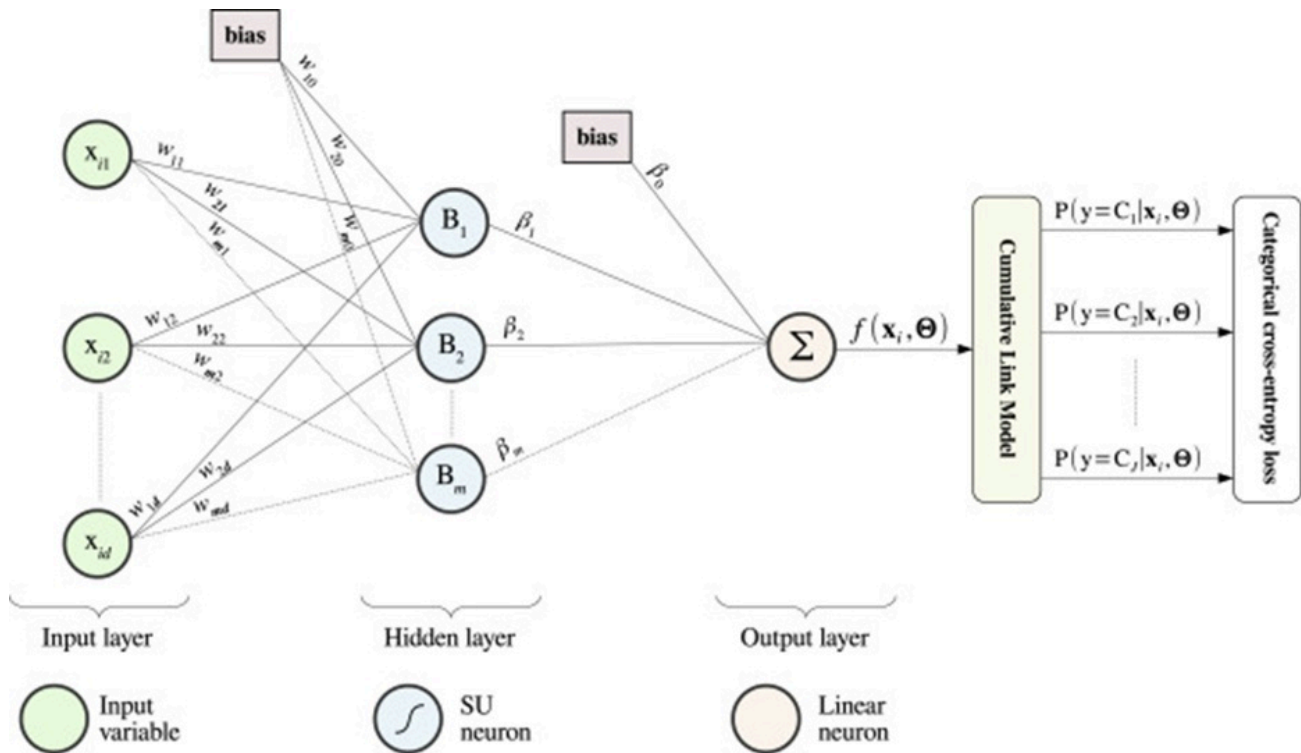


Figure 1. Architecture of the proposed MLP-CLMO approach.

The mechanism by which MLP-CLMO generates ordinal predictions using the CLM attached to the output neuron is described below. CLMs estimate the probabilities of cumulative groups of consecutive classes along an ordinal scale, relying on a latent variable. In the case of MLP-CLMO, this latent variable corresponds to the projection produced by the output neuron, $f(x_i, \Theta)$. These probabilities represent cumulative probabilities $P(y \leq C_j | x_i, \Theta)$ and are defined as follows:

$$P(y \preceq \mathcal{C}_j | \mathbf{x}_i, \Theta) = g(b_j - f(\mathbf{x}_i, \Theta)), \quad (3)$$

Where $g(\cdot)$ denotes the probit link function [26], and $b^T = \{b_0, b_1, b_2, \dots, b_j, \dots, b_J\}$ represents the vector of thresholds used to partition the latent variable projection into J ordinal classes. As the thresholds are required to be strictly increasing in ordinal problems, they are expressed in the following form:

$$b_j = b_1 + \sum_{i=2}^j \gamma_i^2, \quad j = 2, \dots, J-1, \quad (4)$$

Where b_1 is learned to determine the first threshold, $\gamma = \{\gamma_2, \gamma_3, \dots, \gamma_{J-1}\}$ is learned to calculate the subsequent thresholds, and $b_0 = -\infty$, $b_J = +\infty$. This parameterization ensures that $b_0 \leq b_1 \leq \dots \leq b_J$, a necessary condition for the cumulative probabilities $P(y \leq C_j | x_i, \Theta)$ to be non-decreasing as j increases.

These cumulative probabilities can be directly related to the standard (non-cumulative) class probabilities. Accordingly, the probability that the true class y_i of instance x_i is C_j is computed by the CLM in the following manner:

$$P(y = \mathcal{C}_1 | \mathbf{x}_i, \Theta) = P(y \preceq \mathcal{C}_1 | \mathbf{x}_i, \Theta), \quad (5)$$

$$P(y = \mathcal{C}_j | \mathbf{x}_i, \Theta) = P(y \preceq \mathcal{C}_j | \mathbf{x}_i, \Theta) - P(y \preceq \mathcal{C}_{j-1} | \mathbf{x}_i, \Theta), \quad j = 2, \dots, J-1, \quad (6)$$

$$P(y = \mathcal{C}_J | \mathbf{x}_i, \Theta) = 1 - P(y \preceq \mathcal{C}_{J-1} | \mathbf{x}_i, \Theta), \quad (7)$$

And MLP-CLMO assigns the predicted class by selecting $\operatorname{argmax}_{C \in \{1, 2, \dots, J\}} P(y=C | x_i, \Theta)$.

To illustrate the mechanism more clearly, **Figure 2** depicts a representative example of how MLP-CLMO assigns probabilities to each class using the latent projection and the threshold vector. The x-axis displays the values of the projections $f(x_i, \Theta)$ generated by the MLP component, and each curve—corresponding to one class—indicates the probability that an instance projected to that particular x-position belongs to the respective class. It should be noted that the precise shape of these probability curves is governed by both the selected link function and the specific thresholds obtained during training. For this illustration, the probit link function has been applied with a chosen set of thresholds.

In the training process for MLP-CLMO, the model is optimized by minimizing the Categorical Cross-Entropy (CCE) loss, which is formulated as:

$$\begin{aligned} \mathcal{L}(\mathbf{X}, \mathbf{y}, \Theta) \\ = \sum_{i=1}^N \sum_{j=1}^J [p_j(y_i) \cdot (-\log P(y = \mathcal{C}_j | \mathbf{x}_i, \Theta))] \end{aligned} \quad (8)$$

Where $p_j(y_i)$ represents the probability assigned to the i th instance belonging to class C_j , under the hard-label framework, this value equals 1 if $y_i = C_j$ and 0 for all other classes. Finally, $P(y = C_j | x_i, \Theta)$ corresponds to the estimated probability—computed by the model under evaluation—that the instance x_i belongs to class C_j .

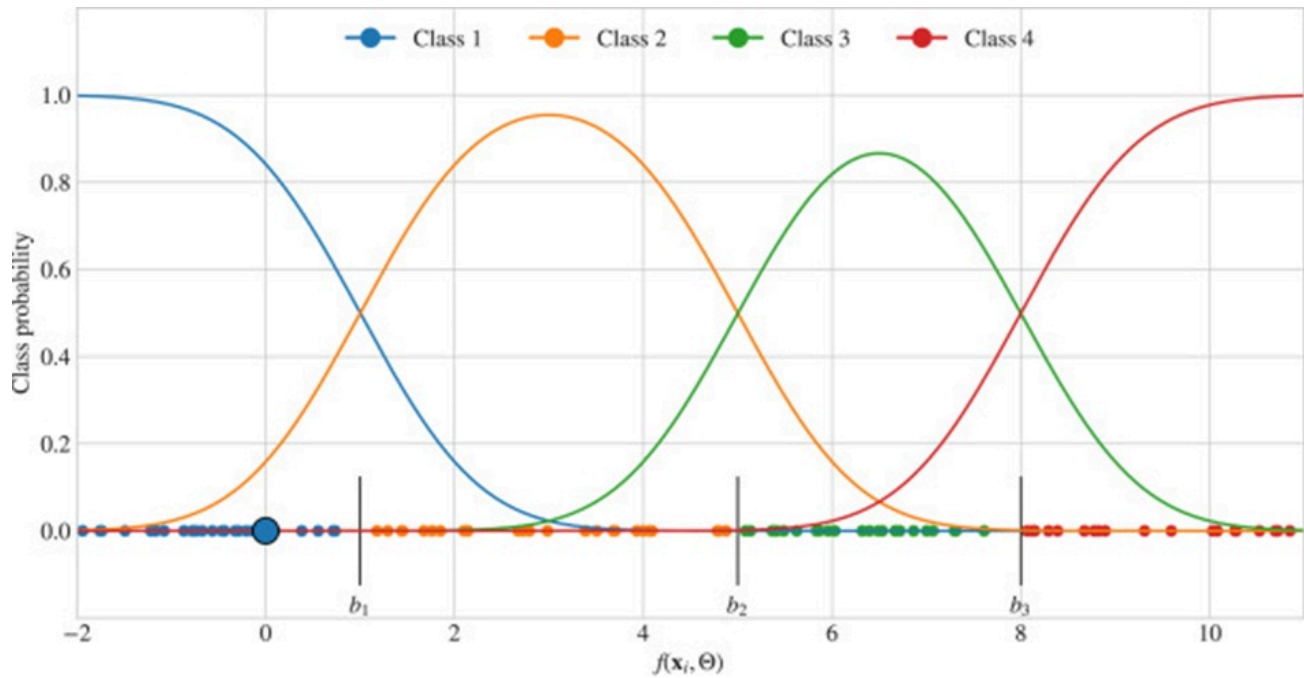


Figure 2. Demonstration of how the MLP-CLMO approach operates on an ordinal classification task involving 4 classes. The horizontal axis displays the patterns mapped onto a continuous real-valued line (obtained from the final linear layer of the MLP, denoted $f(x_i, \Theta)$). The threshold vector $b^T = (b_1, \dots, b_3)$ defines the boundaries that divide the classes. Each dot is colored to indicate which class it belongs to, based on its position relative to the thresholds. The smooth curves illustrate the probability density for each class across all projected values. As an example, at the projection position $x=0$, the probability for each class is given by the height of the corresponding curve at that x -value. Here, the bulk of the probability is concentrated on class C1 (represented by the blue curve), a smaller amount falls on C2 (orange curve), and the other classes are assigned zero probability. (For interpretation of the color references appearing in this figure caption, please consult the online version of the article.)

Soft labeling based on triangular distributions

Soft labeling [27, 32] involves substituting conventional hard (one-hot) labels with continuous soft label distributions. In a classification setting with J classes, the standard hard label for class C_j takes the form of a probability vector p that can be written as:

$$\mathbf{p} = \langle p_1, p_2, \dots, p_j, \dots, p_J \rangle, \quad p_j \in \{0, 1\}, \quad (9)$$

$$\sum_{j=1}^J p_j = 1,$$

Where each element p_j represents the probability that a given instance belongs to class C_j , assuming the instance carries that specific label; as a result, exactly one component equals 1. In contrast, all the others are set to 0.

A soft label can likewise be understood as a probability vector distributed across the J classes. Unlike hard labels, however, the individual components here are allowed to take any value between 0 and 1 inclusive, which permits a much richer and more nuanced representation of the label. Accordingly, a soft label is expressed as:

$$\mathbf{p}^* = \left\langle p_1^*, p_2^*, \dots, p_j^*, \dots, p_J^* \right\rangle, \quad p_j^* \in [0, 1], \quad (10)$$

$$\sum_{j=1}^J p_j^* = 1.$$

This method proves particularly beneficial when addressing the intrinsic uncertainty typical of real-world ordinal classification problems, where distinguishing between instances from neighboring classes is especially challenging owing to their high degree of similarity. Moreover, the approach serves as a form of regularisation, mitigating the impact of noisy labels and reducing the risk of overfitting.

Consequently, incorporating soft labeling into the loss function used during model training enables more effective handling of this uncertainty, which, in turn, can lead to enhanced model performance. This modification can be implemented in the Categorical Cross-Entropy (CCE) loss defined in Eq. 8 by substituting the original hard labels $p_j(y_i)$ with the corresponding soft labels $p_j^*(y_i)$:

$$\mathcal{L}(\mathbf{X}, \mathbf{y}, \Theta)$$

$$= \sum_{i=1}^N \sum_{j=1}^J \left[p_j^*(y_i) \cdot (-\log P(y = \mathcal{C}_j | \mathbf{x}_i, \Theta)) \right], \quad (11)$$

Where $p_j^*(y_i)$ corresponds to the j th element of the soft label linked to the true class of the i th instance, soft labels may be constructed by merging the original hard labels with a chosen probability distribution, according to the following expression:

$$p_j^*(y_i) = (1 - \eta) p_j(y_i) + \eta P(y = \mathcal{C}_j | y_i), \quad (12)$$

η is a parameter ranging from [0,1] that regulates how much smoothing is applied to the labels: when $\eta = 0$, there is no smoothing whatsoever, while $\eta = 1$ produces labels that are fully smoothed. Lastly, $P(y = \mathcal{C}_j | y_i)$ denotes the probability that pattern i belongs to class j , assuming y_i is its true (actual) label.

To determine the probabilities $P(y = \mathcal{C}_j | y_i)$ appearing in Eq. 12, unimodal probability distributions turn out to be especially appropriate for ordinal problems. Their peak (mode) or central tendency (mean) can be deliberately located at the middle of an intermediate class interval or else at one of the extremes (upper or lower bound) for the boundary classes, which allows for a more faithful estimation of the model's loss [32]. It is equally important that the distribution has low variance and concentrates the majority of its probability mass within the interval associated with the true class, with the density falling off rapidly as one moves farther away. Such a shape promotes prediction mistakes mostly in neighboring classes instead of far-removed ones — precisely the desired behavior in ordinal classification tasks. Earlier literature suggested using discrete distributions such as the Binomial [33] or the Poisson. These choices, however, have an important shortcoming: they do not provide sufficient flexibility to position the mode exactly at the center of a class interval while maintaining a sufficiently small variance. The continuous triangular distribution [27], by comparison, brings a clear advantage over those discrete alternatives because it naturally minimizes variance and requires estimating only one additional parameter. This single tunable parameter, called α , directly controls the likelihood of assigning probability to immediately adjacent classes.

Consequently, triangular distributions have been chosen here to model the soft-label probability for every class in the studied problem. The specific parameter settings required to construct each triangular distribution are computed from α , as outlined in the cited reference [27]. An illustrative set of such distributions for a four-class scenario is shown in **Figure 3**, which shows how soft labels are derived for each class.

This ordinal approach is likewise implemented using a multilayer perceptron (MLP) neural network; its overall structure is shown in **Figure 4**.

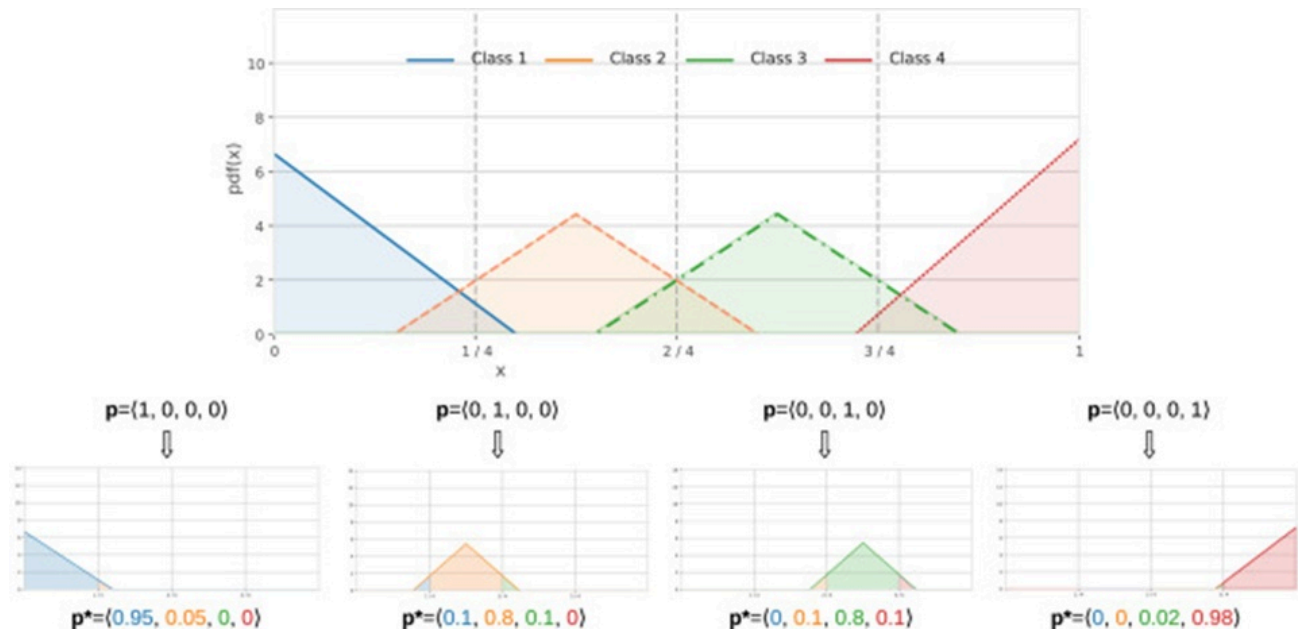


Figure 3. Illustration of the continuous triangular probability distributions utilized within the MLP-TO framework to generate soft labels. A separate distribution is assigned to each of the 4 classes to produce the appropriate soft label. Each panel displays the probability density function (p.d.f.), and the probability allocated to a given class equals the area beneath its distribution curve over that class's defined interval.

Observe that the input and hidden layers in MLP-TO match those of MLP-CLMO. The output layer, however, contains exactly J neurons — one dedicated to each problem class. Adopting the same notational convention as in **Eq. 1**, the activation of the j th output neuron for input pattern i in MLP-TO is given by:

$$f_j(x_i, \Theta_j) = f(x_i, W, \beta_j) = \beta_0 j + \sum_{l=1}^M \beta_l j B_l(x_i, w_l), \quad j=1, \dots, J \quad (13)$$

in which $\Theta_j = (W, \beta_j)$ and the vector $\beta_j^T = (\beta_0 j, \beta_1 j, \dots, \beta_m j)$ gathers all connection weights (including the bias $\beta_0 j$) from the hidden layer to the j th output unit.

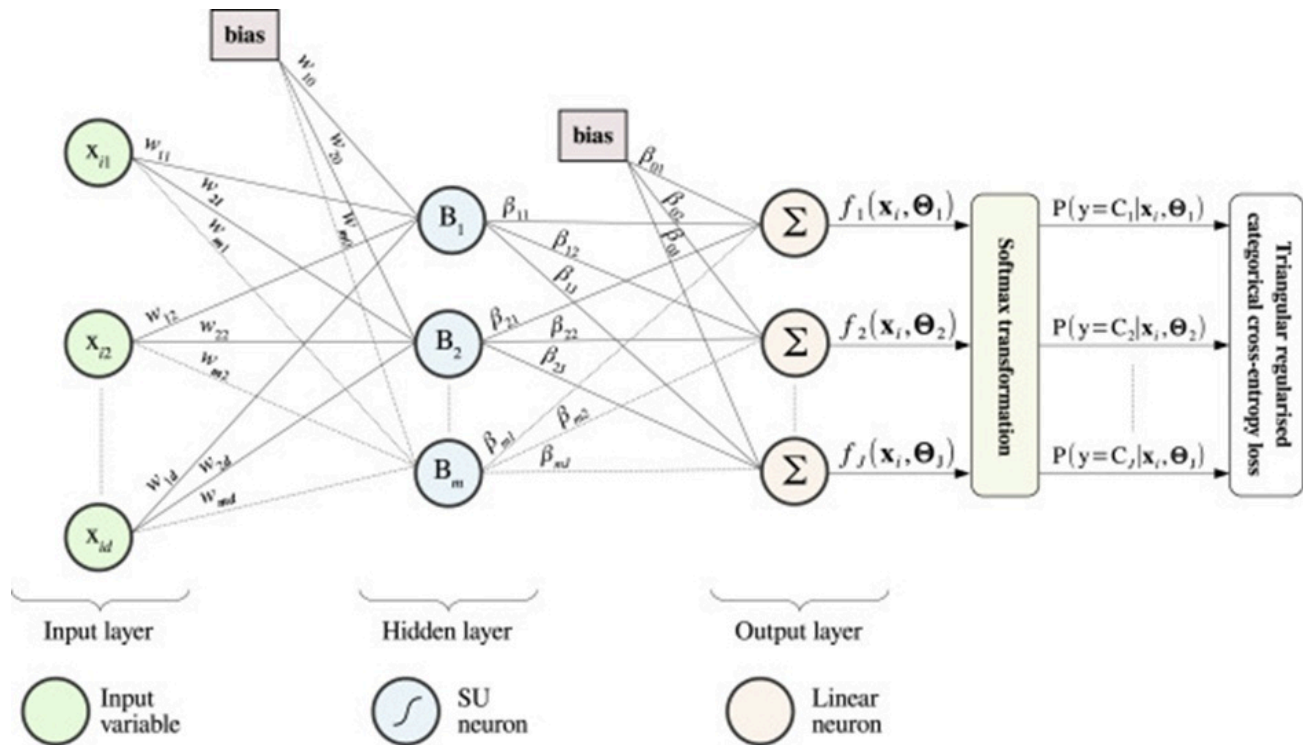


Figure 4. Schematic diagram of the proposed MLP-TO architecture.

Ordinal class prediction is achieved by first converting each MLP-TO output value (Eq. 13) into a class probability via the softmax function:

$$g_j(\mathbf{x}_i, \Theta_j) = \frac{\exp(f_j(\mathbf{x}_i, \Theta_j))}{\sum_{k=1}^J \exp(f_k(\mathbf{x}_i, \Theta_k))}, \quad (0 \leq g_j(\mathbf{x}_i, \Theta_j) \leq 1), \quad (14)$$

$$j = 1, \dots, J,$$

Where $g_j(\mathbf{x}_i, \Theta_j)$ expresses the estimated probability that instance i belongs to class C_j , the ultimate predicted label is then obtained by selecting the class index that maximizes the probability: $\text{argmax}_{j \in \{1, 2, \dots, J\}} g_j(\mathbf{x}_i, \Theta_j)$.

Dataset Description and Formulation of the Ordinal Classification Task

The dataset comprises real hourly wind-speed time series (longer than 10 years) recorded at a wind farm situated in Salamanca, Spain (coordinates 40.79N, 5.49W).

Records span November 1995 through February 2013. After discarding entries with missing wind-speed values, the cleaned dataset contains 133,110 valid samples. Wind speeds are discretized into four ordered classes:

$$y_i = \begin{cases} \mathcal{C}_1, & v_i \in [0, 6] \text{ m/s}, \\ \mathcal{C}_2, & v_i \in (6, 12] \text{ m/s}, \\ \mathcal{C}_3, & v_i \in (12, 20] \text{ m/s}, \\ \mathcal{C}_4, & v_i > 20 \text{ m/s}, \end{cases} \quad (15)$$

Where v_i indicates the wind speed value for the i th sample.

Table 1 reports the sample counts per class for the one-hour-ahead forecasting ($t + 1$ h horizon). Class C1 reflects calm or very low wind conditions, resulting in zero power generation, whereas class C4 captures dangerously high wind speeds that force turbine shutdown to prevent mechanical failure. Both extremes, therefore, correspond to periods of minimal (or null) wind-farm output. Special attention is given to the rare and critically important class C4 (extreme winds), which suffers from severe class imbalance and is consequently very hard to predict accurately. Nevertheless, reliable detection of C4 is vital, since such events can inflict substantial structural damage and cause prolonged production losses.

Table 1. Distribution of training and test samples across classes for wind-speed forecasting at the ($t + 1$ h) horizon.

Class	Train set (103,920 samples)	Test set (29,190 samples)
C1	43,972 samples (42.31%)	12,771 samples (43.75%)
C2	39,753 samples (38.25%)	10,935 samples (37.46%)
C3	17,696 samples (17.03%)	4,996 samples (17.12%)
C4	2,499 samples (2.41%)	488 samples (1.67%)

The input features consist of six variables extracted from the ERA5 reanalysis dataset [34], supplemented by the observed wind speed at the current time t and at the four preceding hours ($t-1$, $t-2$, $t-3$, $t-4$). Forecasts are produced for three different horizons: 1 h, 4 h, and 8 h ahead (i.e., target times $t+1$ h, $t+4$ h, $t+8$ h). The complete initial feature set includes multiple ERA5 fields — among them u - and v -wind components at various altitudes, 2-meter air temperature, and mean sea-level pressure — totaling 35 predictors. A preliminary feature-selection stage reduced this set to the 20 most informative variables. Selection was performed using the Neural Network Evolutionary Programming (NNEP) algorithm [35], an evolutionary method that simultaneously optimizes neural network topology and parameters while performing implicit feature selection. NNEP achieves this through targeted structural mutations: adding/removing links, adding/removing neurons, and neuron fusion [35]. Thirty independent NNEP runs were executed, adopting parameter values close to those recommended in [35] with slight modifications to better suit the ordinal character and difficulty of the wind-speed task. The final selected features were those that appeared in at least 75% of the 30 best-performing models across all runs.

Results and Discussion

This section outlines the experimental configuration, encompassing the evaluation metrics selected and the comparative baseline approaches used to benchmark the two proposed methods. To facilitate reproducibility, all code used to run the experiments has been made openly accessible on GitHub.² The obtained performance outcomes are presented here as well, placing special focus on the accurate forecasting of extreme wind speed conditions. Lastly, a statistical examination of the findings is provided.

Performance measures

In ordinal classification scenarios, selecting suitable evaluation metrics that properly reflect the ordered structure of the classes is essential. Errors between neighboring classes warrant smaller penalties than those between classes that are far apart, which calls for error-weighting schemes that differentiate by mistake severity. Accordingly, the following three metrics are utilized in this study:

- Average Mean Absolute Error (AMAE) [36]: This measure incorporates both the ordinal ranking of true and predicted classes and the extent of class imbalance present in the dataset. AMAE was introduced specifically to overcome shortcomings of the standard Mean Absolute Error (MAE) when applied to imbalanced data. It computes a macro-averaged version of the per-class MAE values — that is, it computes the MAE for each class separately and then averages them equally across classes. AMAE scores range from $[0, J-1]$, with lower values indicating superior model performance. It is formally expressed as follows:

$$\text{AMAE} = \frac{1}{J} \sum_{j=1}^J \text{MAE}_j \in [0, J-1], \quad (16)$$

Where J is the number of classes, and MAE_j is computed as:

$$\begin{aligned} \text{MAE}_j &= \frac{1}{N_j} \sum_{i=1}^{N_j} |\mathcal{O}(y_i) - \mathcal{O}(\hat{y}_i)| \\ &\in [0, J-1], \end{aligned} \quad (17)$$

Where N_j denotes the number of samples belonging to class j , and y_i and $\hat{y}_i$ represent the true (observed) class and the predicted class for pattern i , respectively. • Quadratic Weighted Kappa (QWK) [37]. This metric extends Cohen's Kappa coefficient and quantifies the level of agreement between the actual and predicted classifications. QWK scores generally fall within the interval $[0,1]$, where larger values reflect stronger agreement (and thus better model performance). That said, when the observed agreement is worse than what would be expected under random classification, QWK can take negative values in the range $[-1,0]$. The formal definition of the QWK metric is given as follows:

$$\text{QWK} = 1 - \frac{\sum_{i=1}^N \sum_{j=1}^N \Omega_{ij} O_{ij}}{\sum_{i=1}^N \sum_{j=1}^N \Omega_{ij} E_{ij}} \in [-1, 1], \quad (18)$$

Where Ω denotes the penalty (or weighting) matrix, whose entries are determined based on the discrepancy between the true and predicted class labels. In the quadratic weighting scheme, each matrix element is calculated as $\Omega_{ij} = (i-j)^2 / (J-1)^2 \in [0,1]$. E represents the matrix of expected agreement scores under chance, with each entry defined as $E_{ij} = (O_{i \cdot} \times O_{\cdot j}) / N$, and O stands for the confusion matrix, where $O_{i,j}$ indicates the count of instances from actual class C_i that were assigned to predicted class C_j . The row marginal $O_{i \cdot} = \sum_{k=1}^K O_{i,k}$ gives the total number of patterns from class i , while the column marginal $O_{\cdot j} = \sum_{k=1}^K O_{k,j}$ gives the total number of patterns predicted as class j . • Balanced Accuracy (BA) [38]: BA serves as a highly suitable evaluation metric for imbalanced classification problems, such as the one addressed in this study. It is computed as the average of the per-class recall (sensitivity) values and produces scores in the interval $[0,1]$, with higher values indicating better performance. More precisely, it is expressed as:

$$BA = \frac{\sum_{j=1}^J \frac{o_{jj}}{o_{j\bullet}}}{J} \in [0, 1]. \quad (19)$$

It should be noted that the selection of these three evaluation metrics collectively addresses the two primary considerations in this study: (1) the inherently ordered character of the classification task; and (2) the pronounced class imbalance within the dataset. With respect to the second aspect, particular importance is attached to achieving reliable predictions for the most extreme class — namely, class C4, corresponding to severe wind speeds exceeding 20 m/s. Consequently, in addition to the overall metrics, the effectiveness of both the newly proposed methods and the established benchmark techniques is further evaluated by examining the per-class sensitivity for this critical category, denoted S_4 , which is formally defined as follows:

$$S_4 = \frac{o_{4,4}}{o_{j\bullet}} \in [0, 1]. \quad (20)$$

Benchmark methods and hyperparameter optimization

The two novel methods introduced — MLP-CLMO and MLP-TO — are evaluated in comparison with several well-established nominal and ordinal classification techniques reported in the literature:

- LRN: standard logistic regression.
- MLPN: a conventional multilayer perceptron featuring a softmax output layer and trained via the categorical cross-entropy loss.
- LROIT [28]: ordinal logistic regression based on the immediate-thresholds formulation.
- LROAT [28]: ordinal logistic regression based on the all-thresholds formulation.

To improve clarity, nominal classification models are labeled with the superscript N, whereas ordinal classification models are labeled with the superscript O. Among these baselines, MLPN is treated as an ablation study for the proposed approaches. The primary distinction between MLPN and MLP-CLMO lies in their prediction mechanisms: MLPN uses softmax probabilities, while MLP-CLMO employs the cumulative link model (CLM). In contrast, the main difference between MLPN and MLP-TO concerns the loss function applied during training: categorical cross-entropy is used for MLPN, whereas MLP-TO utilizes a triangular-regularised cross-entropy loss. Furthermore, linear models are included because of their proven effectiveness on large-scale datasets. Accordingly, we adopt the classic LRN, along with its two ordinal extensions, LROIT and LROAT. We also incorporate the persistence model [39] as a simple yet strong baseline; this naive method consistently forecasts the next value to be identical to the most recent observed value and frequently yields competitive performance.

A uniform training protocol is applied to both proposed methods (MLP-CLMO and MLP-TO) and to all comparative state-of-the-art techniques. Hyperparameter optimization is performed via randomized search on the training set, employing 3-fold cross-validation. The search space for each method's hyperparameters is defined in [Table 2](#). Up to 50 iterations are conducted across the 3 folds, guided by the AMAE metric. Once the optimal hyperparameter set is identified through cross-validation, the final model is retrained on the complete training dataset. It should be highlighted that MLPN, serving

as an ablation variant, shares the same hyperparameter search ranges as MLP-CLMO and MLP-TO for all common tunable parameters.

Table 2. Ranges of hyperparameter values explored during optimization for each method.

Technique(s)	Parameter description	Range of values
LRN, LROAT, LROIT	Regularisation strength	$\{10^{-3}, 10^{-2}, \dots, 10^3\}$
MLPN	Number of hidden neurons	$\{10, 25, 50, 100, 150, 200\}$
	Initial learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$
	Number of iterations	$\{500, 1000, 1500, 2000, 2500, 3000\}$
MLP-CLMO	Number of hidden neurons	$\{10, 25, 50, 100, 150, 200\}$
	Initial learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$
	Number of iterations	$\{500, 1000, 1500, 2000, 2500, 3000\}$
MLP-TO	Number of hidden neurons	$\{10, 25, 50, 100, 150, 200\}$
	Initial learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$
	Number of iterations	$\{500, 1000, 1500, 2000, 2500, 3000\}$
	Upper bound of error in adjacent classes (α)	$\{0.03, 0.04, 0.05\}$
	Label softness in regularisation (η)	$\{0.8, 0.9, 1.0\}$

Experimental results

This section reports the performance of the proposed MLP-CLMO and MLP-TO models, along with a thorough comparison with the four benchmark techniques and the persistence baseline. Results are presented according to the three evaluation metrics: AMAE, QWK, and BA (Tables 3–5). AMAE and QWK are particularly well-suited for assessing ordinal classifiers because they explicitly account for the ordered nature of the classes. Meanwhile, BA is especially appropriate for measuring performance under class imbalance.

For the non-deterministic methods (MLPN, MLP-CLMO, and MLP-TO), performance is summarised as the mean $\pm$ standard deviation (Mean $\pm$ SD) across 30 independent runs. Each run uses a unique random seed, and the same sequence of seeds is applied across all three stochastic models to guarantee equitable comparisons. The first three columns display results for each forecast horizon (1 h, 4 h, and 8 h). In comparison, the final two columns report the overall average across all horizons and the average rank, respectively.

The initial analysis focuses on AMAE — the metric that guided hyperparameter tuning. As shown in Table 3, MLP-CLMO delivers the strongest performance across all three prediction horizons. It also records the best overall average (0.357) and the top average rank (1.000). The next-best outcomes are produced by MLPN, which secures second place across all horizons (1 h, 4 h, and 8 h), though it ties with MLP-TO at the 1 h horizon. MLPN likewise achieves the second-best average score (0.365) and average rank (2.333) across horizons, followed closely by MLP-TO.

Regarding QWK (Table 4), MLP-CLMO yields the highest scores in two of the three horizons (4 h and 8 h). It also attains the best average value (0.717) and the best average rank (2.333) when aggregated across all horizons. MLPN again ranks second overall, posting the second-highest QWK scores for every horizon and the second-best average. For the 1 h horizon specifically, the top performers are LRN, LROAT, and LROIT.

Table 3. AMAE results ($\downarrow$). The top-performing value is shown in bold, and the second-best value is shown in italics.

Classifier	1-hour horizon	4-hour horizon	8-hour horizon	Mean error	Mean rank
Persistence	0.196	0.464	0.669	0.443	4.667
LRN	0.199	0.485	0.702	0.462	6.333
MLPN	0.152 $\pm$ 0.001	0.382 $\pm$ 0.002	0.562 $\pm$ 0.010	0.365	2.333
LROAT	0.191	0.483	0.710	0.462	5.667
LROIT	0.191	0.478	0.750	0.473	5.333
MLP-CLMO	0.151 $\pm$ 0.001	0.375 $\pm$ 0.002	0.545 $\pm$ 0.004	0.357	1.000
MLP-TO	0.152 $\pm$ 0.001	0.385 $\pm$ 0.002	0.566 $\pm$ 0.006	0.368	2.667

Additionally, the Balanced Accuracy (BA) results in **Table 5** show that the leading method remains the proposed MLP-CLMO, with the proposed MLP-TO in close second place. MLP-CLMO delivers the highest BA scores across every prediction horizon, securing the strongest overall average (0.678) and the top average rank (1.000). Notably, at the 1 h horizon, MLP-CLMO and MLP-TO tie for first place. For the initial two horizons (1 h and 4 h), MLPN records the second-highest results, while MLP-TO takes second place at the 8 h horizon. Overall, MLP-TO matches MLPN for the second-best average score (0.675) and shares the same second-best average rank (2.333).

Table 4. Results achieved in terms of QWK ($\uparrow$). The best result is highlighted in bold, whereas the second best is in italics.

Classifier	1-hour horizon	4-hour horizon	8-hour horizon	Mean error	Mean rank
Persistence	0.869	0.671	0.503	0.681	6.667
LRN	0.875	0.686	0.528	0.696	4.000
MLPN	0.873 $\pm$ 0.000	0.705 $\pm$ 0.001	0.549 $\pm$ 0.005	0.709	2.667
LROAT	0.875	0.683	0.523	0.694	4.000
LROIT	0.875	0.687	0.488	0.683	4.333
MLP-CLMO	0.872 $\pm$ 0.000	0.711 $\pm$ 0.001	0.569 $\pm$ 0.002	0.717	2.333
MLP-TO	0.871 $\pm$ 0.000	0.703 $\pm$ 0.002	0.544 $\pm$ 0.004	0.706	4.000

Taken together, these findings demonstrate the clear advantage of the newly proposed methods over both established state-of-the-art techniques and the persistence baseline. Examining the persistence model's performance, it performs reasonably well at the shortest (1 h) horizon. However, as the forecast lead time lengthens, its accuracy declines markedly and fails to remain competitive with the other methods. A similar degradation with increasing horizon is evident in several benchmark approaches, including the nominal LRN model and its ordinal variants LROAT and LROIT. Although MLPN delivers respectable results, it is consistently surpassed by its ordinal counterpart, MLP-CLMO, which leverages the cumulative link model (CLM) for ordinal output. Meanwhile, MLP-TO closely tracks the strong performance pattern set by MLP-CLMO and produces outstanding BA scores.

Table 5. Results achieved in terms of BA ($\uparrow$). The best result is highlighted in bold, whereas the second best is in italics.

Classifier	1-hour horizon	4-hour horizon	8-hour horizon	Mean error	Mean rank
------------	----------------	----------------	----------------	------------	-----------

Persistence	0.805	0.568	0.446	0.606	4.667
LRN	0.801	0.544	0.413	0.586	6.667
MLPN	0.848 ± 0.001	0.643 ± 0.002	0.534 ± 0.006	0.675	2.667
LROAT	0.809	0.546	0.407	0.587	6.000
LROIT	0.809	0.553	0.416	0.593	4.667
MLP-CLMO	0.849 ± 0.001	0.649 ± 0.002	0.537 ± 0.003	0.678	1.000
MLP-TO	0.849 ± 0.001	0.641 ± 0.002	0.536 ± 0.004	0.675	2.333

It is also worth noting that, across the board, the two ordinal logistic regression models — LROAT and LROIT — generally surpass their nominal counterpart, LRN, on both AMAE and BA, while achieving equivalent performance on QWK.

Prediction of extreme wind speeds

The evaluations conducted so far account for the dataset's imbalance and ordinal characteristics using the BA, AMAE, and QWK metrics. Still, they do not isolate performance on the rare and critical extreme class C4. In this application, assessing global model quality across all classes is valuable, but accurately forecasting extreme wind speeds (i.e., those exceeding 20 m/s, class C4) is particularly practical. To address this, the sensitivity specific to class C4, denoted S4, is computed for each method. The corresponding results appear in [Table 6](#). As shown, MLP-TO surpasses all other approaches at all three prediction horizons, with its margin of improvement increasing at the longer horizons (4 h and 8 h). On average across horizons, MLP-TO correctly detects extreme wind speed events in 94.4% of cases at the 1 h horizon, 77.6% at the 4 h horizon, and 71.8% at the 8 h horizon — a still highly competitive figure considering the extended forecast window. In aggregate, MLP-TO records the highest mean S4 value (0.813) and the best average rank (1.000) over all horizons.

Table 6. Results achieved in terms of S4 (†). The best result is highlighted in bold, whereas the second best is in italics.

Classifier	1-hour horizon	4-hour horizon	8-hour horizon	Mean error	Mean rank
Persistence	0.719	0.385	0.240	0.448	4.167
LRN	0.682	0.241	0.038	0.321	7.000
MLPN	0.922 ± 0.003	0.732 ± 0.009	0.664 ± 0.019	0.773	3.000
LROAT	0.717	0.274	0.053	0.348	6.000
LROIT	0.719	0.296	0.103	0.373	4.833
MLP-CLMO	0.934 ± 0.003	0.759 ± 0.007	0.670 ± 0.010	0.787	2.000
MLP-TO	0.944 ± 0.004	0.776 ± 0.010	0.718 ± 0.029	0.813	1.000

Statistical validation

This part summarises the hypothesis-testing procedures used to determine whether the apparent differences in average performance — measured by AMAE, BA, QWK, and S4 — across stochastic models and across forecast horizons are statistically significant.

To begin, Kolmogorov–Smirnov tests were run on sets of 30 results for each metric, horizon, and stochastic method. These tests checked compliance with a normal distribution. At the conventional threshold of $\alpha = 0.05$, the normality

assumption (null hypothesis) was upheld in every single case. Accordingly, it can be reasonably assumed that the distributions of all performance scores, irrespective of method or horizon, are normal.

Levene's tests were then performed to evaluate homogeneity of variance across the same groupings (metric $\times$ horizon $\times$ method). Again, with $\alpha = 0.05$, the null hypothesis of equal variances was not rejected in any instance.

With both normality and homoscedasticity confirmed, parametric procedures were justified for comparing mean performances. One-way analysis of variance (ANOVA) was therefore executed independently for each prediction horizon and each of the four metrics. In every test, highly significant differences emerged among the three stochastic methods (all P-values < 0.001 at $\alpha = 0.05$).

Because these ANOVA results indicated meaningful variation among the methods for each metric, follow-up pairwise comparisons were conducted using Tukey's Honestly Significant Difference (HSD) procedure. The detailed pairwise outcomes for AMAE, BA, QWK, and S4 appear in [Tables 7–10](#), respectively. In each table, methods assigned to the same homogeneous subset show statistically indistinguishable performance, whereas methods placed in separate subsets differ at a statistically significant level.

With respect to AMAE ([Table 7](#)):

- At the 1 h horizon, MLP-CLMO stands alone in the superior subset and is significantly better than the remaining methods (which share one lower subset).
- At the 4 h horizon, the three methods fall into three distinct subsets, confirming significant differences among all pairs, and MLP-CLMO ranks first.
- At the 8 h horizon, the pattern repeats that of the 1 h case: MLP-CLMO is uniquely placed in the best-performing subset.

Table 7. Results of the post-hoc HSD Tukey's test for the AMAE (↓) metric.

Model/Measure	Forecast Horizon: 1 h		Forecast horizon: 4 h			Forecast horizon: 8 h	
	1	2	1	2	3	1	2
MLP-CLMO	0.1512	—	0.3747	—	—	0.5450	—
MLPN	—	0.1520	0.3816	—	—	—	0.5621
MLP-TO	—	0.1519	—	0.3854	—	—	0.5658
p-values	1.000	0.916	1.000	1.000	1.000	1.000	0.120

Concerning BA ([Table 8](#)):

- For the 1 h horizon, MLP-CLMO occupies its own top subset as the undisputed leader; MLP-TO ranks next, followed by MLPN, with no significant separation between the latter two.
- At the 4 h horizon, each method belongs to a separate subset (all pairwise differences significant), MLP-CLMO leads, and MLPN shows the weakest result.
- For the 8 h horizon, MLP-CLMO again secures the highest position and shares the leading subset; here, MLP-TO is not significantly different from either MLP-CLMO or MLPN, while only MLPN differs significantly from MLP-CLMO.

Table 8. Results of the post-hoc HSD Tukey's test for the BA (↑) metric.

Model/Measure	Forecast horizon: 1 h		Forecast horizon: 4 h			Forecast horizon: 8 h	
	1	2	1	2	3	1	2
MLPN	0.8485	—	0.6430	—	—	0.5342	—
MLP-TO	0.8486	—	—	0.6411	—	0.5356	0.5356
MLP-CLMO	—	0.8493	—	0.6487	—	—	0.5373
p-values	0.788	1.000	1.000	1.000	1.000	0.422	0.347

For QWK (Table 9), clear statistical differences exist among the three methods at all horizons:

- MLP-CLMO outperforms the others at both the 4 h and 8 h horizons.
- At the 1 h horizon, however, MLPN records the strongest result.

Table 9. Results of the post-hoc HSD Tukey's test for the QWK ($\uparrow$) metric.

Model/Metric	Prediction horizon: 1 h			Prediction horizon: 4 h			Prediction horizon: 8 h		
	1	2	3	1	2	3	1	2	3
MLP-TO	0.8714	—	—	0.7032	—	—	0.5441	—	—
MLPN	—	—	0.8728	—	0.7054	—	—	0.5493	—
MLP-CLMO	—	0.8719	—	—	0.7108	—	—	0.5690	—
p-values	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Finally, regarding sensitivity to the extreme class, S4 (Table 10):

- The pattern is consistent for the 1 h and 4 h horizons: all three methods differ significantly from one another, and MLP-TO consistently achieves the highest score.
- The same ranking and significance pattern holds at the 8 h horizon, where MLP-TO again forms its own top subset, distinctly outperforming the other two methods.

Table 10. Results of the post-hoc HSD Tukey's test for the S4 ($\uparrow$) metric.

Model/Measure	Forecast horizon: 1 h			Forecast horizon: 4 h			Forecast horizon: 8 h	
	1	2	3	1	2	3	1	2
MLPN	0.9216	—	—	0.7315	—	—	0.6644	—
MLP-CLMO	—	0.9342	—	—	0.7586	—	0.6696	—
MLP-TO	—	—	0.9440	—	0.7761	—	—	0.7179
p-values	1.000	1.000	1.000	1.000	1.000	1.000	0.602	1.000

Overall, the combination of metric values and statistical test results indicates that MLP-CLMO provides the most robust performance on AMAE, BA, and QWK. At the same time, MLP-TO demonstrates clear superiority, specifically in correctly identifying instances of the extreme wind-speed class (S4).

This section provides a deeper examination of the experimental findings through multiple visualization approaches. First, the performance distributions produced by the stochastic methods are inspected across the four evaluation metrics. Then, attention turns specifically to the two proposed models — MLP-CLMO and MLP-TO — by visually comparing their predicted class labels with the observed classes over selected time intervals with strong fluctuations in wind speed, for each of the three forecast horizons (1 h, 4 h, and 8 h). In addition, confusion matrices are shown for both models across all three prediction horizons. Lastly, the practical operational suitability of the two proposed methods for real-world implementation in wind farm control systems is evaluated.

Distribution analysis of results from stochastic methods

To provide a complete overview of the outcomes across the four performance indicators, **Figure 5** illustrates the distribution of results from the stochastic approaches for AMAE, QWK, BA, and S4. Each distribution is based on 30 independent runs. Importantly, the y-axis scale is adjusted individually for each sub-panel to maximize readability; a uniform scale across all plots would have made visual interpretation much more difficult due to differing value ranges. As shown in the figure, the distributions are tightly grouped in nearly all cases, highlighting the high reliability and consistency of the three competing classifiers. Broadly speaking, predictive performance deteriorates as the forecast horizon lengthens. The most noticeable drop occurs between the shorter horizons (1 h and 4 h) and the longest one (8 h), a pattern that reflects the intrinsically greater difficulty of long-range forecasting and the resulting increase in classification mistakes. This degradation tends to be especially apparent in AMAE and BA, since both metrics are particularly sensitive to class imbalance — a prominent feature of the wind speed classification task examined here. Furthermore, the performance gaps among the methods generally widen as the horizon length increases. For example, on AMAE, the proposed MLP-CLMO consistently delivers outstanding results and clearly outperforms the other two models; however, at the 8 h horizon, the distinction between MLP-TO and the baseline MLPN becomes much less pronounced. A similar pattern emerges for BA, though the advantage of MLP-TO over MLPN is somewhat more evident even at longer horizons. In the case of QWK, the shortest horizon (1 h) shows that the conventional MLPN delivers the strongest performance, surpassing the proposed MLP-CLMO; however, this advantage reverses at the medium and long horizons (4 h and 8 h), where MLP-CLMO takes the lead. Finally, the S4 metric (sensitivity to the extreme class) most clearly showcases the strength of the proposed MLP-TO, which reliably achieves the highest scores and outperforms both MLPN and MLP-CLMO across all horizons.

Taken together, these visualizations confirm the overall superiority of the two ordinal classification approaches introduced in this work, with their advantages becoming increasingly evident at longer prediction horizons. For this reason, both MLP-CLMO and MLP-TO are selected for the more detailed follow-up analyses presented in the subsequent subsections.

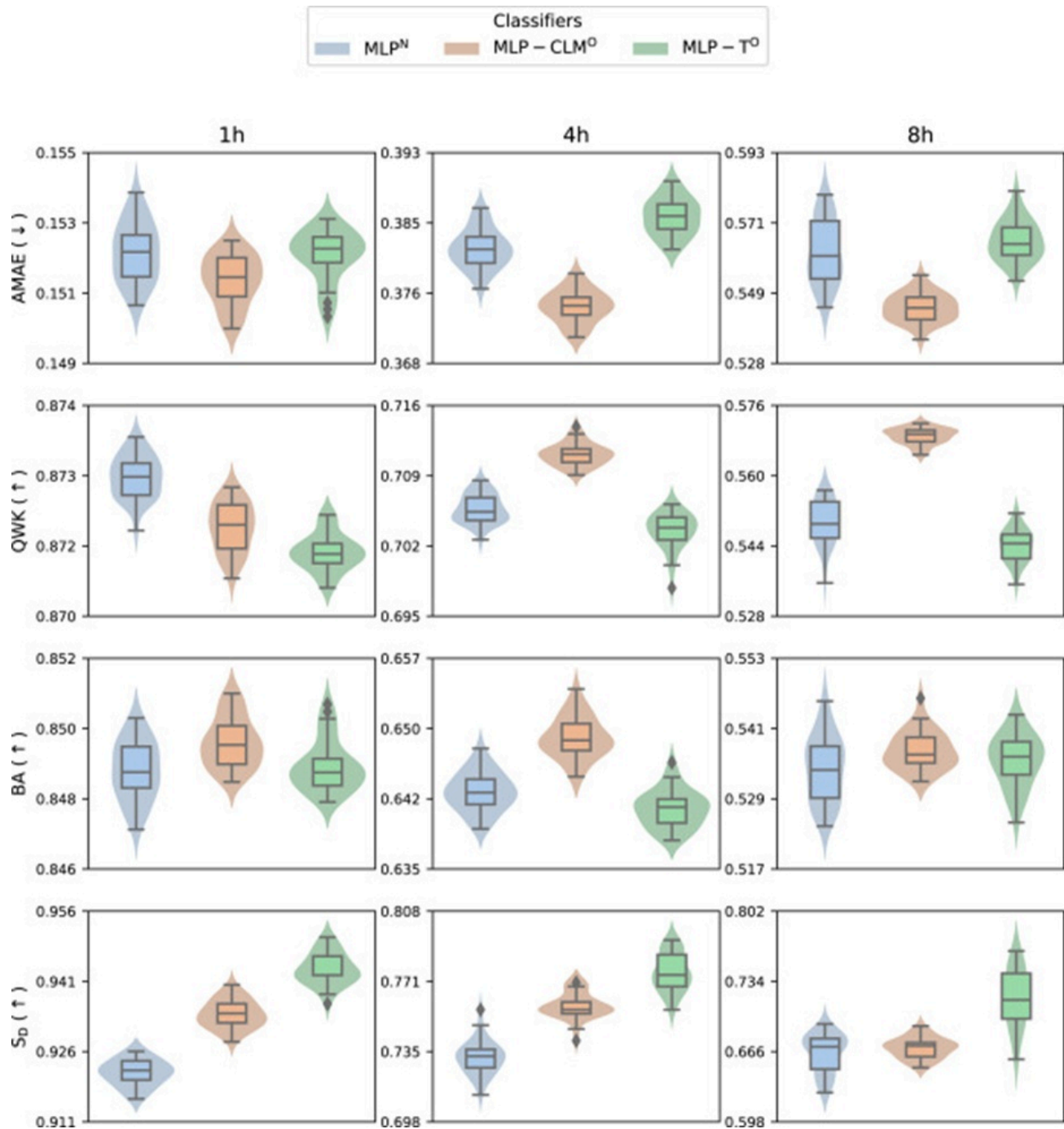


Figure 5. Spread of performance values obtained by each MLP-based method across the four evaluation metrics considered in the study.

Evaluation of predictive performance through visual comparison

Figures 6 and 7 illustrate the degree of trustworthiness exhibited by the MLP-CLMO and MLP-TO approaches, respectively. Their central objective is to juxtapose the forecasts from the two ordinal classification models against the recorded measurements. All three forecasting intervals (1 h, 4 h, and 8 h) are evaluated over a full month, during which the test portions of the datasets exhibit substantial fluctuations. To ensure fairness and remove run-specific effects, only the outcomes from the initial execution of each algorithm are shown. Across the board, the most accurate results are obtained at the shortest horizon (1 h), where both ordinal models exhibit very few errors. The MLP-CLMO and MLP-TO

forecasts successfully replicate the predominant direction and shape of the time series; whenever an incorrect class is assigned, the mistake almost always involves a neighboring category. Even at extended horizons, the two classifiers manage to reproduce the principal movement of the series. That said, short-lived rapid variations—for instance, the swings recorded from 25 February 2009 through 1 March 2009—prove progressively harder to anticipate accurately as the look-ahead period grows. Notably, the MLP-CLMO variant stands out for its ability to detect sharp, discontinuous jumps in wind speed, including the pronounced shifts visible near 14 February 2009.

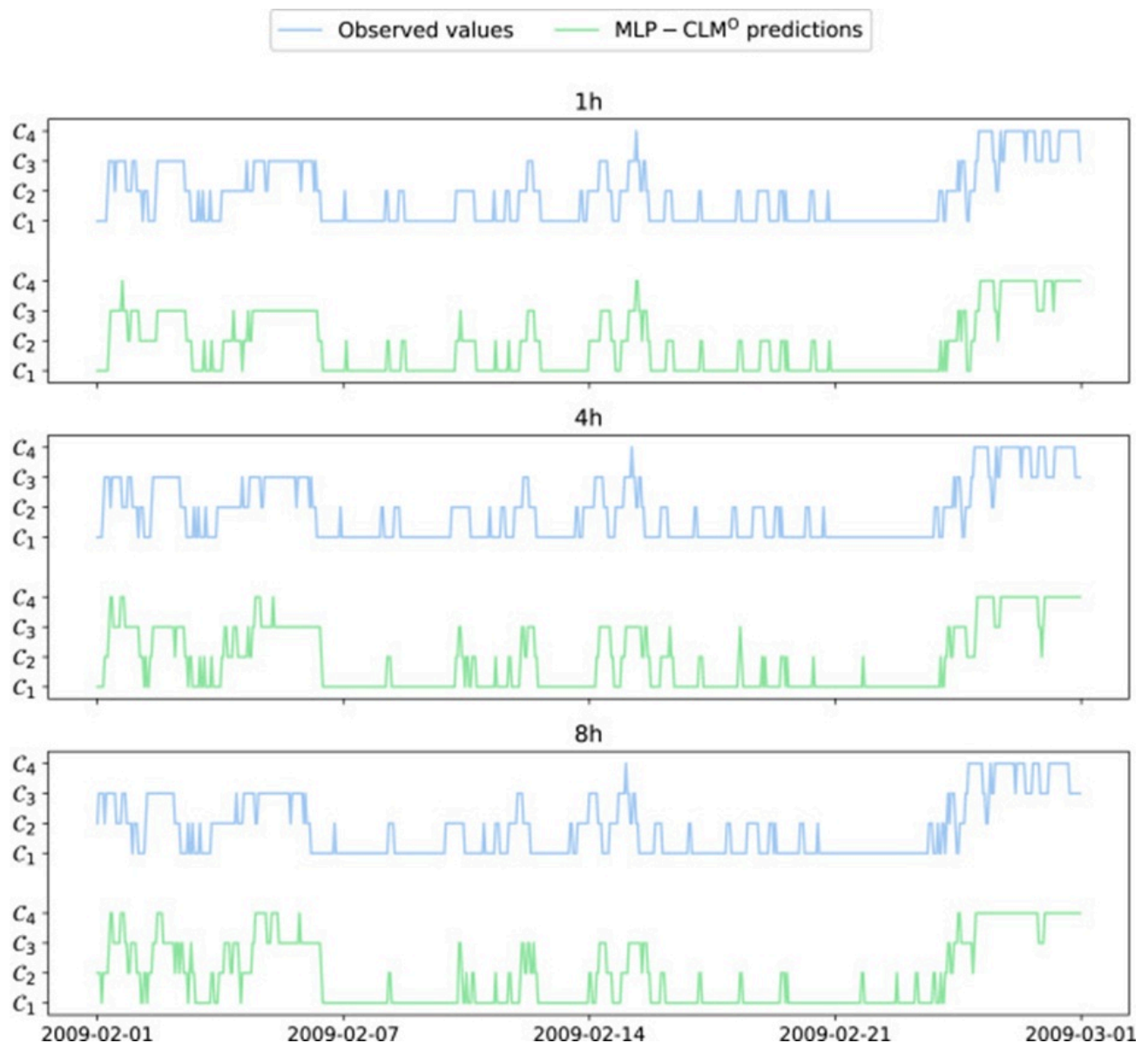


Figure 6. Observed wind speed values contrasted with forecasts generated by the MLP-CLMO model at the three different horizons.

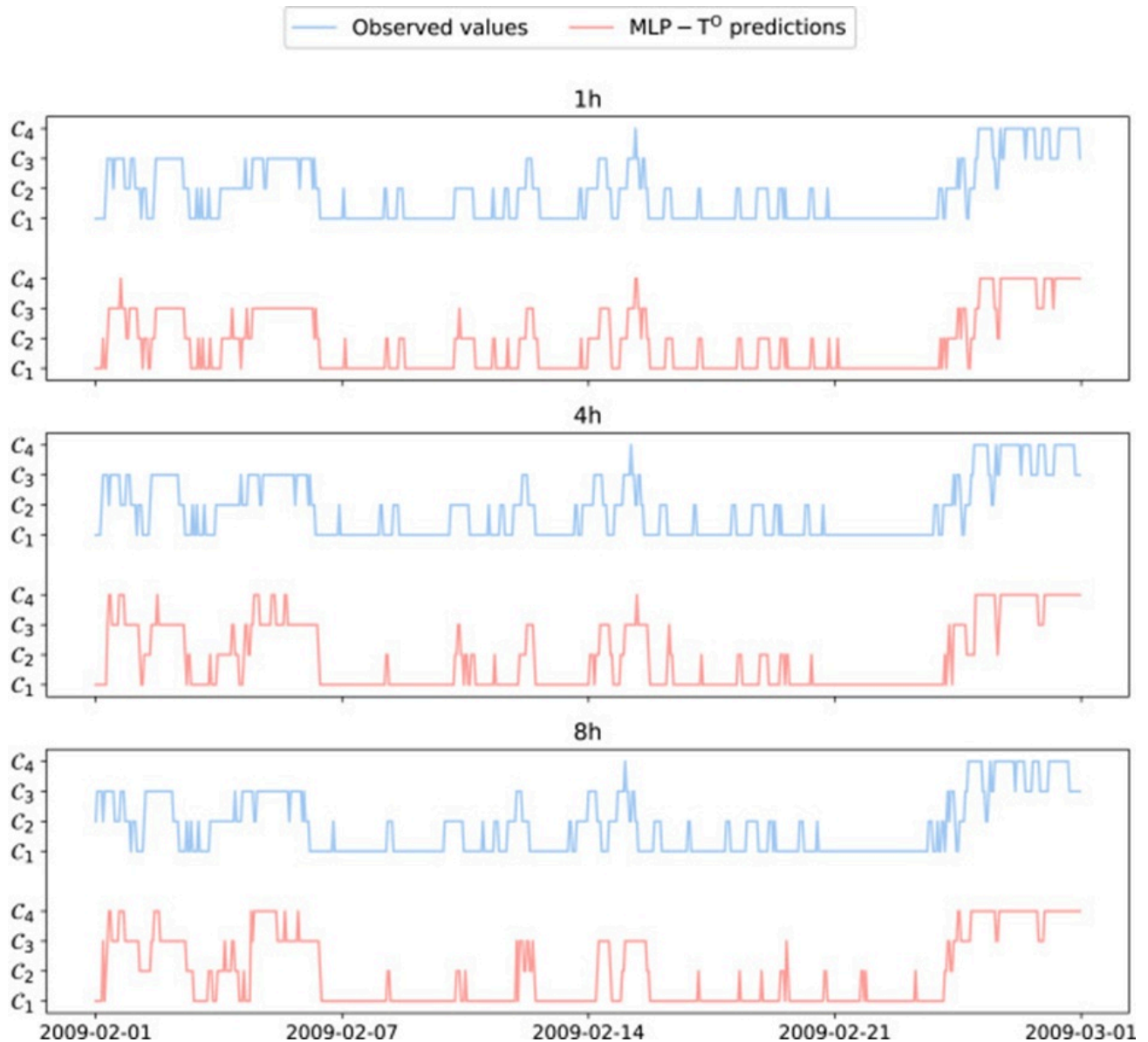


Figure 7. Observed wind speed values contrasted with forecasts generated by the MLP-TO model at the three different horizons.

Detailed inspection of confusion matrices for both models

The present section compiles the confusion matrices computed for MLP-CLMO and MLP-TO over each of the three forecasting distances. These matrices supply a supplementary graphical perspective on model effectiveness and pinpoint precisely which categories tend to be confused. For every combination of model and horizon, two matrices appear in **Figure 8**: one reflecting the single best random seed according to the AMAE score (rendered in blue) and another showing the averaged result over all 30 independent executions (rendered in magenta). Color intensity in both gradients is proportional to the quantity of samples belonging to each wind-speed category.

Across every horizon, the averaged (magenta) matrix for MLP-CLMO deviates only modestly from its best-run counterpart (blue). The MLP-TO model exhibits behavior analogous to that of the MLP-TO model, confirming that both methods remain remarkably consistent and dependable across prediction lengths. At the 1-hour horizon, the overwhelming proportion of samples falls on the main diagonal for both algorithms, meaning correct assignments dominate; the

occasional errors are virtually always restricted to immediately adjacent wind classes. A comparable situation emerges at the 4-hour horizon—albeit with a predictable rise in overall error count—yet misassignments continue to concentrate in neighboring classes rather than remote ones. Such a pattern confirms that the classifiers appropriately exploit the ordered structure of the target variable and avoid large jumps in predicted category. Extending to the 8-hour case, correct classifications still constitute the majority, although class C2 stands out as noticeably more problematic. Because this represents the toughest forecasting challenge, misclassification frequency naturally climbs relative to the 1 h and 4 h tasks. Crucially, however, the large majority of those errors remain localized to adjacent classes—an outcome that is highly desirable in ordinal settings.

To conclude, MLP-CLMO demonstrates superior fidelity in representing the complete distribution of classes, whereas MLP-TO consistently delivers stronger results on the tail-end categories (especially extremes). These distinct profiles become particularly evident when examining class C4, which captures the most severe wind-speed episodes. As horizon lengthens, MLP-TO progressively outperforms MLP-CLMO on this critical extreme class.

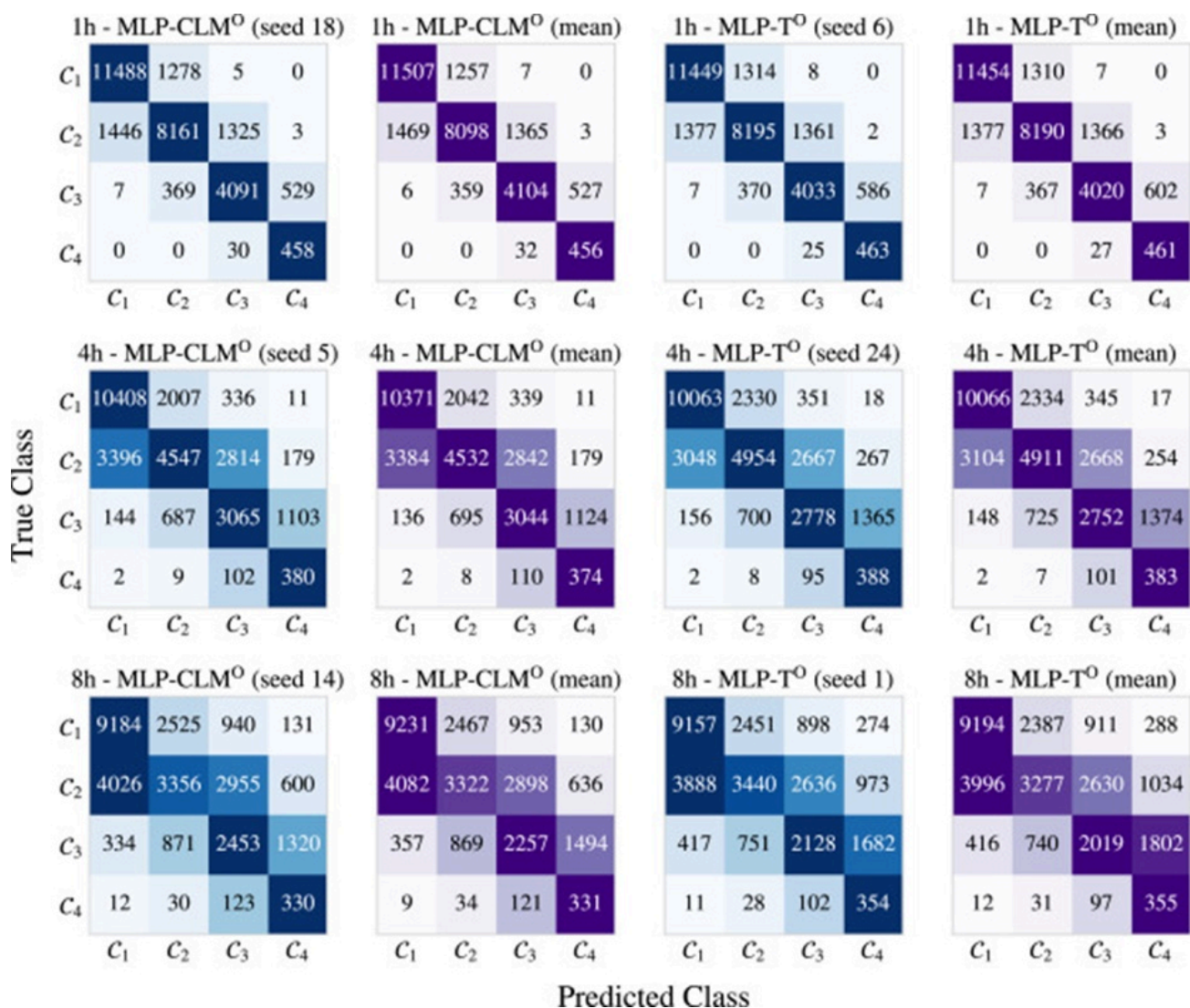


Figure 8. Confusion matrices obtained with MLP-CLMO and MLP-TO for the 1 h, 4 h, and 8 h wind-speed prediction tasks. Matrices in blue depict the single best-performing seed judged by the AMAE criterion. Matrices in magenta represent the average across the 30 separate runs, so that class counts may differ marginally from those in the corresponding best matrix. Gradient shading (blue and magenta) indicates the total number of instances per class (For interpretation of the color references appearing in the figure caption, please consult the online version of the article).

Assessment of computational efficiency for MLP-CLMO and MLP-TO

This final section examines the practical deployability and resource demands of the MLP-CLMO and MLP-TO methods when integrated into operational wind farm environments. In this regard, **Table 11** summarises key indicators, including the number of hidden-layer neurons, the number of training iterations, the durations required for cross-validation (C-V) and final training, and the computational load expressed in Millions of Floating-point Operations Per Second (MFLOPS).

From the table, the hidden-layer neuron count ranges from 50 to 200 for the MLP-CLMO method and from 100 to 200 for the MLP-TO method. Cross-validation durations range from 83.38 to 598.47 minutes for MLP-CLMO and from 276.20 to 684.86 minutes for MLP-TO. It should be emphasized that these C-V times encompass a total of 151 model fittings (which can be executed in parallel): namely, 50 distinct hyperparameter sets evaluated via 3-fold cross-validation, plus one additional full training run using the optimal hyperparameter set identified. Regarding final training time alone, values range from 0.55 to 3.96 minutes for MLP-CLMO and from 1.83 to 4.54 minutes for MLP-TO. Inference times are omitted from the table since they remain insignificant (under one second in all cases). The observed variations in execution times stem primarily from differences in the number of hidden neurons and the number of iterations selected during cross-validation. Moreover, because all experiments were conducted on a shared, heterogeneous computing cluster featuring diverse processors and memory configurations, disparities in timing—whether between the two methods or across prediction horizons within the same method—should not be over-interpreted. Execution duration is heavily influenced by the specific processor assigned to each job, as well as by concurrent network activity and disk I/O contention on the cluster. Consequently, the lightweight, shallow architectures employed in both MLP-CLMO and MLP-TO eliminate the need for specialized hardware accelerators, yielding acceptably short training periods and virtually instantaneous inference. These characteristics underscore the models' strong computational efficiency and strongly support their practical applicability within real-world wind farm control systems. This efficiency is further corroborated by the modest complexity figures reported in Millions of Floating-point Operations Per Second (MFLOPS), which translate to very low overall resource utilization.

Table 11. Evaluation of operational efficiency for the MLP-CLMO and MLP-TO approaches across the 1 h, 4 h, and 8 h prediction horizons, detailing the number of hidden neurons, number of iterations, cross-validation (C-V) time, training time, and complexity measured in Millions of Floating-point Operations Per Second (MFLOPS). Cross-validation time covers a total of 151 model fittings (parallelizable): 50 hyperparameter configurations × 3 folds + one final training pass with the selected best hyperparameter set. Training time reflects only the final optimization run across the determined number of iterations. MFLOPS complexity corresponds to the operations required to generate a single wind-speed prediction for one input sample. Rows labeled “Best” report values for the single run that achieved the lowest AMAE metric, while rows labeled “Mean” present averages computed over the 30 independent runs conducted.

Prediction horizon	Result type	Approach	Hidden neurons	Iterations	Cross-validation time (min)	Training time (min)	Model complexity (MFLOPS)
1 h	Best (seed 18)	MLP-CLMO	200	2500	598.47	3.96	0.005
	Mean	MLP-CLMO	185	2167	535.21	3.54	0.005
	Best (seed 6)	MLP-TO	200	1500	684.86	4.54	0.005
	Mean	MLP-TO	160	1850	496.52	3.29	0.004
4 h	Best (seed 5)	MLP-CLMO	50	2000	174.66	1.16	0.001

	Mean	MLP-CLMO	85	1867	291.57	1.93	0.002
	Best (seed 24)	MLP-TO	100	3000	465.62	3.08	0.003
	Mean	MLP-TO	148	1950	276.20	1.83	0.004
8 h	Best (seed 14)	MLP-CLMO	150	1000	83.38	0.55	0.004
	Mean	MLP-CLMO	72	1683	245.97	1.63	0.002
	Best (seed 1)	MLP-TO	150	1000	501.16	3.32	0.004
	Mean	MLP-TO	148	1450	345.05	2.29	0.004

Conclusion

The present study proposes and rigorously evaluates two innovative ordinal classification frameworks for wind speed forecasting at wind farm sites, each using four distinct ordinal categories. These categories effectively summarise operationally relevant wind regimes: speeds corresponding to negligible or zero power output (below 6 m/s), moderate generation levels (6–12 m/s), high production ranges (12–20 m/s), and extreme conditions that either maximize output or trigger turbine cut-out for safety reasons (above 25 m/s). A key characteristic of these categories is their inherent ordinal structure, which renders conventional nominal classifiers less appropriate and favors the adoption of dedicated ordinal techniques. The core aim of the research was to benchmark two state-of-the-art ordinal classifiers against the multi-horizon wind speed prediction problem, with forecasts generated separately for three distinct look-ahead periods: 1 h, 4 h, and 8 h. Beyond pursuing strong global classification performance, particular emphasis was placed on the reliable detection of the most critical (extreme) class associated with severe wind events.

The target variable consisted of hourly wind speed measurements recorded at a medium-scale Spanish wind farm from November 1995 to February 2013. This continuous variable was discretized into the four previously defined ordinal classes using the specified speed thresholds. As input features, six meteorological variables—temperature, mean sea level pressure, dew point, among others—were extracted from the ERA5 Reanalysis dataset. Together with the wind farm's own historical wind speed observations, data from the five preceding hourly timestamps were initially included for the 1-h horizon, yielding a preliminary set of 35 predictors. A feature selection procedure was subsequently applied, retaining only the 20 most influential variables. It should be noted that all three resulting datasets (one per prediction horizon) exhibited pronounced class imbalance, particularly due to the very limited representation of the extreme wind class.

The first proposed method, designated MLP-CLMO, is based on an Artificial Neural Network (ANN) architecture that incorporates the Cumulative Link Model (CLM) to perform ordinal prediction. This formulation prioritizes balanced, high-quality performance distributed across the entire class range. The second method, named MLP-TO, employs an ANN trained via a soft-labeling strategy grounded in triangular probability distributions. This technique proves especially effective when confronted with severe class imbalance, as encountered in the present datasets.

For benchmarking purposes, an extensive suite of competitive approaches was included, encompassing various nominal and ordinal linear models, together with the simple persistence forecast as a reference. One additional baseline, termed

MLPN, served as an ablation study to isolate the contributions of the core components in MLP-CLMO and MLP-TO. Model performance was quantified using four complementary evaluation metrics: Average Mean Absolute Error (AMAE), Quadratic Weighted Kappa (QWK), Balanced Accuracy (BA), and S4 (sensitivity to the extreme class). Empirical results consistently showed that both proposed methods outperformed all reference techniques across every metric considered. In particular, MLP-CLMO delivered statistically superior results on the AMAE, QWK, and BA indicators, effectively addressing both the ordinal structure and the data imbalance.

In contrast, MLP-TO exhibited markedly better performance on the S4 metric, demonstrating its specialized capability to improve extreme-class detection while maintaining competitive accuracy on the remaining classes. Moreover, the advantage of the two proposed methods became increasingly pronounced as the forecasting horizon lengthened (particularly at 4 h and 8 h). Statistical significance testing of the randomized algorithms confirmed meaningful differences: MLP-CLMO significantly excelled in AMAE, BA, and QWK, whereas MLP-TO significantly outperformed competitors in terms of S4. Lastly, an evaluation of computational practicality confirmed that both models exhibit short training durations, near-instantaneous inference, and very low resource demands (measured in MFLOPS), rendering them highly suitable for real-time integration into wind farm operational systems.

Looking ahead, future investigations will focus on integrating explainable artificial intelligence (XAI) techniques into this application domain. The application of XAI is expected to offer valuable insights into the dominant drivers of extreme wind speed predictions. In particular, we intend to identify the most influential meteorological and temporal factors driving extreme events, with the ultimate objective of improving early warning capabilities and overall forecast reliability for these high-impact scenarios.

Acknowledgements

This study has been partially supported by the “Agencia Estatal de Investigación (España)”, Spanish Ministry of Science, Innovation and Universities (grant refs.: [PID2023-150663NB-C21](#) and [PID2023-150663NB-C22/AEI/10.13039/501100011033](#)), by the European Commission, projects “CLImate INTElligence: Extreme events detection, attribution and adaptation design using machine learning, CLINT” (grant ref.: [H2020-LC-CLA-2020-2, 101003876](#)) and “Test and Experiment Facilities for the Agri-Food Domain, AgriFoodTEF” (grant ref.: [DIGITAL-2022-CLOUD-AI-02, 101100622](#)), by the ENIA International Chair in Agriculture, University of Córdoba (grant ref.: [TSI-100921-2023-3](#)), by the European Union through the European Regional Development Fund; “project IA-CONV” funded by the Community of Madrid through the grant agreement for the promotion and advancement of research and technology transfer at the University of Alcalá (grant ref.: [CM/DEMG/2024-039](#)), and by the University of Córdoba through competitive grants for Andalusian society challenges (grant ref.: [PP2F_L1_15](#)). Antonio Manuel Gómez-Orellana has been supported by “Consejería de Transformación Económica, Industria, Conocimiento y Universidades de la Junta de Andalucía” (grant ref.: [PREDOC-00489](#)). David Guijo-Rubio has been supported by the “Agencia Estatal de Investigación (España)” [MCIU/AEI/10.13039/501100011033](#) and European Union NextGenerationEU/PRTR (grant ref.: [JDC2022-048378-I](#)).

Conflict of interest

None

Financial support

None

Ethics statement

None

Received: 12 Oct 2021

Revised: 30 Oct 2021

Accepted: 18 Nov 2021

Published online: 20 March 2022

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

- Mantesi E, Chmutina K, Goodier C. The office of the future: Operational energy consumption in the post-pandemic era. *Energy Res Soc Sci*. 2022;87:102472.
<https://doi.org/10.1016/j.erss.2021.102472>.
- Wang Q, Zhang F. What does China's economic recovery after COVID-19 pandemic mean for the economic growth and energy consumption of other countries? *J Clean Prod*. 2021;295:126265.
<https://doi.org/10.1016/j.jclepro.2021.126265>.
- Rowland CS, Mjelde JW. Politics and petroleum: Unintended implications of global oil demand reduction policies. *Energy Res Soc Sci*. 2016;11:209-24.
<https://doi.org/10.1016/j.erss.2015.10.003>.
- Welsby D, Price J, Pye S, Ekins P. Unextractable fossil fuels in a 1.5°C world. *Nature*. 2021;597(7875):230-4.
<https://doi.org/10.1038/s41586-021-03821-8>.
- Letcher TM. Why wind energy. In: *Wind energy engineering*. Elsevier; 2023:3-10.
<https://doi.org/10.1016/B978-0-323-99353-1.00025-6>.
- Zhang K, Qu Z, Dong Y, Lu H, Leng W, Wang J, et al. Research on a combined model based on linear and nonlinear features - A case study of wind speed forecasting. *Renew Energy*. 2019;130:814-30.
<https://doi.org/10.1016/j.renene.2018.05.093>.
- Lowe R, Drummond P. Solar, wind and logistic substitution in global energy supply to 2050—Barriers and implications. *Renew Sustain Energy Rev*. 2022;153:111720.
<https://doi.org/10.1016/j.rser.2021.111720>.
- Yeh TM, Huang YL. Factors in determining wind farm location: Integrating GQM, fuzzy DEMATEL, and ANP. *Renew Energy*. 2014;66:159-69.
<https://doi.org/10.1016/j.renene.2013.12.003>.
- Santos M, González M. Factors that influence the performance of wind farms. *Renew Energy*. 2019;135:643-51.
<https://doi.org/10.1016/j.renene.2018.12.033>.
- Lu KD, Wu ZG, Huang T. Differential evolution-based three stage dynamic cyber-attack of cyber-physical power systems. *IEEE/ASME Trans Mechatron*. 2022;28(2):1137-48.
<https://doi.org/10.1109/TMECH.2022.3214314>.

Li J, Wang C, Wang H. Deep reinforcement learning for wind and energy storage coordination in wholesale energy and ancillary service markets. *Energy AI*. 2023;14:100280.
<https://doi.org/10.1016/j.egyai.2023.100280>.

Chen MR, Zeng GQ, Lu KD, Weng J. A two-layer nonlinear combination method for short-term wind speed prediction based on ELM, ENN, and LSTM. *IEEE Internet Things J*. 2019;6(4):6997-7010.

Tascikaraoglu A, Uzunoglu M. A review of combined approaches for prediction of short-term wind speed and power. *Renew Sustain Energy Rev*. 2014;34:243-54.
<https://doi.org/10.1016/j.rser.2014.03.033>.

Santhosh M, Venkaiah C, Vinod Kumar D. Current advances and approaches in wind speed and wind power forecasting for improved renewable energy integration: A review. *Eng Rep*. 2020;2(6):e12178.
<https://doi.org/10.1002/eng2.12178>.

Wang J, Song Y, Liu F, Hou R. Analysis and application of forecasting models in wind power integration: A review of multi-step-ahead wind speed forecasting models. *Renew Sustain Energy Rev*. 2016;60:960-81.
<https://doi.org/10.1016/j.rser.2016.01.114>.

Valdivia-Bautista SM, Domínguez-Navarro JA, Pérez-Cisneros M, Vega-Gómez CJ, Castillo-Téllez B. Artificial intelligence in wind speed forecasting: A review. *Energies*. 2023;16(5):2457.
<https://doi.org/10.3390/en16052457>.

Roungkvist JS, Enevoldsen P. Timescale classification in wind forecasting: A review of the state-of-the-art. *J Forecast*. 2020;39(5):757-68.
<https://doi.org/10.1002/for.2657>.

Li W, Chong Y, Guo X, Liu J. A hybrid wind power prediction model based on seasonal feature decomposition and enhanced feature extraction. *Energy AI*. 2024;18:100442.
<https://doi.org/10.1016/j.egyai.2024.100442>.

Wang S, Li B, Li G, Li B, Li H, Jiao K, et al. A comprehensive review on the development of data-driven methods for wind power prediction and AGC performance evaluation in wind-thermal bundled power systems. *Energy AI*. 2024;16:100336.
<https://doi.org/10.1016/j.egyai.2024.100336>.

Pérez-Ortiz M, Jiménez-Fernández S, Gutiérrez PA, Alexandre E, Hervás-Martínez C, Salcedo-Sanz S. A review of classification problems and algorithms in renewable energy applications. *Energies*. 2016;9(8):607.
<https://doi.org/10.3390/en9080607>.

Peláez-Rodríguez C, Pérez-Aracil J, Fister D, Prieto-Godino L, Deo R, Salcedo-Sanz S. A hierarchical classification/regression algorithm for improving extreme wind speed events prediction. *Renew Energy*. 2022;201:157-78.
<https://doi.org/10.1016/j.renene.2022.11.042>.

Gutiérrez PA, Salcedo-Sanz S, Hervás-Martínez C, Carro-Calvo L, Sánchez-Monedero J, Prieto L. Evaluating nominal and ordinal classifiers for wind speed prediction from synoptic pressure patterns. In: 2011 11th International Conference on Intelligent Systems Design and Applications. IEEE; 2011:1265-70.

Georgoulas G, Kolios S, Karvelis P, Stylios C. Examining nominal and ordinal classifiers for forecasting wind speed. In: 2016 IEEE 8th International Conference on Intelligent Systems (IS). IEEE; 2016. p. 504-9.
<https://doi.org/10.1109/IS.2016.7737469>.

Gutiérrez PA, Salcedo-Sanz S, Hervás-Martínez C, Carro-Calvo L, Sánchez-Monedero J, Prieto L. Ordinal and nominal classification of wind speed from synoptic pressure patterns. *Eng Appl Artif Intell*. 2013;26(3):1008-15.
<https://doi.org/10.1016/j.engappai.2012.10.018>.

Dorado-Moreno M, Gutiérrez PA, Cornejo-Bueno L, Prieto L, Salcedo-Sanz S, Hervás-Martínez C. Ordinal multi-class architecture for predicting wind power ramp events based on reservoir computing. *Neural Process Lett*. 2020;52:57-74.
<https://doi.org/10.1007/s11063-018-9922-5>.

Vargas VM, Gutiérrez PA, Hervás-Martínez C. Cumulative link models for deep ordinal classification. *Neurocomputing*. 2020;401:48-58.
<https://doi.org/10.1016/j.neucom.2020.03.034>.

Vargas VM, Gutiérrez PA, Barbero-Gómez J, Hervás-Martínez C. Soft labelling based on triangular distributions for ordinal classification. *Inf Fusion*. 2023;93:258-67.
<https://doi.org/10.1016/j.inffus.2023.01.003>.

Rennie JD, Srebro N. Loss functions for preference levels: Regression with discrete ordered labels. In: *Proceedings of the IJCAI Multidisciplinary Workshop on Advances in Preference Handling*. Vol 1. AAAI Press; 2005. p. 1-6.

Gutiérrez PA, Pérez-Ortiz M, Sánchez-Monedero J, Fernández-Navarro F, Hervás-Martínez C. Ordinal regression methods: survey and experimental study. *IEEE Trans Knowl Data Eng*. 2015;28(1):127-46.
<https://doi.org/10.1109/TKDE.2015.2457911>.

Gutiérrez PA, García S. Current prospects on ordinal and monotonic classification. *Prog Artif Intell*. 2016;5:171-9.
<https://doi.org/10.1007/s13748-016-0088-y>.

Vargas VM, Gutiérrez PA, Rosati R, Romeo L, Frontoni E, Hervás-Martínez C. Deep learning based hierarchical classifier for weapon stock aesthetic quality control assessment. *Comput Ind*. 2023;144:103786.
<https://doi.org/10.1016/j.compind.2022.103786>.

Vargas VM, Gutiérrez PA, Hervás-Martínez C. Unimodal regularisation based on beta distribution for deep ordinal regression. *Pattern Recognit*. 2022;122:1-10.
<https://doi.org/10.1016/j.patcog.2021.108310>.

Liu X, Fan F, Kong L, Diao Z, Xie W, Lu J, et al. Unimodal regularized neuron stick-breaking for ordinal classification. *Neurocomputing*. 2020;388:34-44.

Hersbach H, Bell B, Berrisford P, Hirahara S, Horányi A, Muñoz-Sabater J, et al. The ERA5 global reanalysis. *Q J R Meteorol Soc*. 2020;146(730):1999-2049.
<https://doi.org/10.1002/qj.3803>.

Amiri B, Gómez-Orellana AM, Gutiérrez PA, Dizène R, Hervás-Martínez C, Dahmani K. A novel approach for global solar irradiation forecasting on tilted plane using Hybrid Evolutionary Neural Networks. *J Clean Prod*. 2021;287:125577.
<https://doi.org/10.1016/j.jclepro.2020.125577>.

Baccianella S, Esuli A, Sebastiani F. Evaluation measures for ordinal regression. In: *2009 Ninth International Conference on Intelligent Systems Design and Applications*. IEEE; 2009. p. 283-7.
<https://doi.org/10.1109/ISDA.2009.230>.

Vargas VM, Gutiérrez PA, Rosati R, Romeo L, Frontoni E, Hervás-Martínez C. Exponential loss regularisation for encouraging ordinal constraint to shotgun stocks quality assessment. *Appl Soft Comput*. 2023;138:110191.
<https://doi.org/10.1016/j.asoc.2023.110191>.

Brodersen KH, Ong CS, Stephan KE, Buhmann JM. The balanced accuracy and its posterior distribution. In: *2010 20th International Conference on Pattern Recognition*. IEEE; 2010. p. 3121-4.
<https://doi.org/10.1109/ICPR.2010.764>.

Salcedo-Sanz S, Casillas-Pérez D, Del Ser J, Casanova-Mateo C, Cuadra L, Piles M, et al. Persistence in complex systems. *Phys Rep*. 2022;957:1-73.

<https://doi.org/10.1016/j.physrep.2022.02.002>.