energies

MDPI

Article

Towards Better Wind Resource Modeling in Complex Terrain: A k-Nearest Neighbors Approach

Pedro Quiroga-Novoa [1], Gabriel Cuevas-Figueroa [1], José Luis Preciado [1], Rogier Floors [2], Alfredo Peña [2] and Oliver Probst [1,*]

[1] School of Engineering and Sciences, Tecnologico de Monterrey, Monterrey, NL CP64689, Mexico; pedroquiroga7@gmail.com (P.Q.-N.); gabriel.cuevas@tec.mx (G.C.-F.); jlpreciadoarreola@tec.mx (J.L.P.)
[2] Wind Energy Department, Technical University of Denmark, 4000 Roskilde, Denmark; rofl@dtu.dk (R.F.); aldi@dtu.dk (A.P.)
* Correspondence: oprobst@tec.mx

Abstract: Wind turbines are often placed in complex terrains, where benefits from orography-related speed up can be capitalized. However, accurately modeling the wind resource over the extended areas covered by a typical wind farm is still challenging over a flat terrain, and over a complex terrain, the challenge can be even be greater. Here, a novel approach for wind resource modeling is proposed, where a linearized flow model is combined with a machine learning approach based on the k-nearest neighbor (*k*-NN) method. Model predictors include combinations of distance, vertical shear exponent, a measure of the terrain complexity and speedup. The method was tested by performing cross-validations on a complex site using the measurements of five tall meteorological towers. All versions of the *k*-NN approach yield significant improvements over the predictions obtained using the linearized model alone; they also outperform the predictions of non-linear flow models. The new method improves the capabilities of current wind resource modeling approaches, and it is easily implemented.

Keywords: wind resource; machine learning; similarity; complex terrain; WAsP; WindSim

Citation: Quiroga-Novoa, P.; Cuevas-Figueroa, G.; Preciado, J.L.; Floors, R.; Peña, A.; Probst, O. Towards Better Wind Resource Modeling in Complex Terrain: A k-Nearest Neighbors Approach. *Energies* **2021**, *14*, 4364. https://doi.org/10.3390/en14144364

Academic Editor: Davide Astolfi and Eugen Rusu

Received: 28 May 2021
Accepted: 15 July 2021
Published: 20 July 2021

1. Introduction

Wind and solar photovoltaic power plants are now the cheapest options for electricity generation in most parts of the world [1], paving the way for the extensive decarbonization [2–4] of the power sector and the economy. However, in the case of wind energy, the accurate assessment of the wind resource at a prospective wind farm site continues to be key to successful development and competitive pricing. While traditional modeling approaches [5,6] for wind project development [7] are often sufficiently accurate in flat or mildly rolling landscapes, wind power predictions in complex terrain are considerably more challenging [6,8–11]. A 5% mean error in predicted wind speed can be the threshold for the approval or rejection of a wind project. Although traditional modeling approaches based on linearized flow solvers are nowadays often complemented with Reynolds-averaged Navier–Stokes (RANS) flow models and sometimes even large-eddy simulations (LES), the benefit is not always clear. Well-tuned linearized flow models, complemented by expert knowledge, may often yield similar or better results compared to advanced computational fluid dynamics (CFD) models.

In an attempt to evaluate and compare the performance of numerical wind flow models, different validation studies have been reported in the literature. Flow modeling approaches include assessments using wind industry-standard linearized flow models such as the one included in the Wind Atlas Analysis and Application Program (WAsP) [5], and more complex CFD approaches, e.g., RANS and LES. Beaucage et al. [6] performed a validation study over different terrain features, where the linearized flow model in WAsP and the Meteodyn CFD model were compared. In contrast to what is normally assumed,

the results of WAsP were similar or better than those of the CFD model, with an overall root mean squared error (RMSE) of 0.62 m/s (corresponding to an 8.0% error) for WAsP and 0.76 m/s (9.4%) for Meteodyn. Another comparative analysis was performed between the linearized WAsP model and the RANS-model also available within the WAsP suite (WAsP CFD) at a complex site in Brazil [8]. In this study, it was pointed out that the WAsP CFD model did not present a clear advantage over the linearized version. The authors conjectured that thermal effects not considered by either model highly contributed to the uncertainty of both results. Other studies have suggested the capacity of CFD models to outperform linear models [9–11]; Hristov et al. [11] suggested improvements of about 8% when predicting the annual energy production (AEP).

More recently, statistical learning methods have made their appearance in wind resource assessment. Such methods have the potential for taking advantage of a larger amount of input data than conventional approaches. In such data-based methods, the learning process commonly relies on terrain and meteorological features to estimate the target variable. This provides greater flexibility for considering micro-climatic effects that are generally not accounted for by flow models. Examples of machine learning methods include ensembles of regression trees [12], support vector regression (SVR) [13] and neural networks (NN) [14].

The combination of physics-based and data-based methods may be a natural solution to the problem of accurate wind resource estimations in a complex terrain. Such hybrid approaches take advantage of each method's strengths to increase robustness and predictive performance. However, for wind resource assessment, these methods are less popular. According to a literature review, only Tang et al. [15] appear to have used a method that combines flow (CFD) simulations with a data-driven technique for assimilating multiple on-site measurements in complex terrain, in addition to a more traditional inverse-distance weighting (IDW) method [16].

Here, we address the continuing challenges of accurate wind resource estimation in complex terrains by designing a method which taps into the capabilities of modern wind resource modeling suites such as WAsP or WindSim, but simultaneously provides the capability of accounting for micro-climatic effects, which are generally not properly addressed in flow models. The proposed new method is based on a conceptually simple machine learning approach, the k-nearest neighbors (*k*-NN) method (see, e.g., [17]). To the best of our knowledge, this approach has not been used before. The basic idea was to take advantage of the *similarity* between different locations at a prospective wind farm site in terms of a set of classifiers or *features* (Section 2.1). Ideally, it should be possible to determine such features with the terrain information alone (elevation and roughness) and possibly wind resource information from one reference location, very much like in conventional flow models. It will be argued below that all feature parameters required for this work can actually be determined with flow models such as WAsP or WindSim alone, although an improved prediction can be obtained in the case of power density estimates if the wind rose at one reference location is known. The new *k*-NN method can then be used in complete analogy to a conventional flow model to predict the wind resource at an arbitrary location for the purpose of turbine yield calculation or wind mapping. It should be noted that while the new method essentially uses the same input information as a conventional flow model, it provides larger flexibility, since it is not restricted by the deterministic relationships between a target and a reference location, which necessarily only consider the properties of the fluid model but not the micro-climate.

In order to build a reference case against which the new method can be compared, flow simulations based on both the linearized model within the WAsP suite (Section 2.6) and the RANS-CFD model (WindSim) (Section 2.7) were conducted first and the results were analyzed by cross-validation. The setup of both models was explored in a detailed manner, ensuring that the models were optimally configured and not artificially underperforming. This included the fine-tuning for atmospheric stability (in the case of WAsP), the use of both standard and high-resolution roughness maps, the exploration of a detailed forest

model (in the case of WindSim), and the study of two different turbulence closure models (in the case of WindSim).

A number of different implementations were studied for the *k*-NN approach (Section 2.4). The basic approach consists of directly using the observed wind speed or power density at the available met tower locations, with the exception of the target location used for validation in each turn. Alternatively, the local wind climates can be first transferred to the target location, and the *k*-NN can then be applied to the transferred climates; this is what we call the *hybrid* approach below. Given that the hybrid method is a statistically independent implementation, the linear combination of both methods, i.e., an *ensemble*, bears the potential of providing more accurate predictions and was implemented as well.

2. Methods and Data

2.1. The k-NN Method Applied to Wind Resource Modeling: The General Concept

The general idea of the *k*-NN approach is to determine similarity between different locations, using a certain number N_f of *features* as variables in a generalized coordinate space. Each site can then be represented as a point in this N_f-dimensional space, and its distance from any other site can be determined by an appropriate norm. Here, the L2-norm $L_2(x) = (\sum x_i^2)^{1/2}$ was used throughout, prior to the standardization of each feature variable x through $x \to (x - \mu_x)/\sigma_x$. Feature candidates include strictly terrain-related variables, such as the terrain complexity parameter RIX [18], speedup and (geometric) distance between sites; variables related to local atmospheric conditions, such as temperature and turbulence intensity; and mixed variables such as the vertical wind shear, which depend on both terrain-dependent flow and atmospheric stability.

The key assumption of the *k*-NN method is that the variable of interest, e.g., the wind speed $v_n = v(t_n)$ at time step t_n, can be predicted from the values of its k nearest neighbors in the N_f-dimensional feature space, either by a simple or a weighted average:

$$\hat{y}(x_0) = \sum_{x_i \in N_k(x_0)} w_i y_i / \sum w_i, \tag{1}$$

where x_0 is the target site and $N_k(x_0)$ is the set of nearest neighbors identified by the algorithm. The weights w_i were taken either as constant (uniform weighting) or as $w_i = 1/d(x_0, x_i)$ (inverse distance weighting, IDW), where $d(x_0, x_i)$ is the distance between features (=$L_2(|x_0 - x_i|)$. It should be noted that the number k of nearest neighbors can be dynamically determined at each time step by the algorithm, allowing to account for time-dependent features.

The *k*-NN procedure is illustrated in Figure 1 for the case of two features, which in this case are the RIX number and the speedup. The differences in RIX between the predictor and the predicted site (delta-RIX) can be used to correct results obtained with the linearized flow model in WAsP for complex terrain analysis [5]. Here, RIX is used as one of the potential features determining similarity. In Figure 1, the target site is shown is a diamond-shaped location in the RIX-speedup plane, together with a number of potential predictor sites. Note that both features may be taken as constant or time dependent; in the more general case of time-dependent features, Figure 1 shows a snapshot of a set of locations at a given time step t_n. The methodology can also be extended to include the *time* as an additional feature.

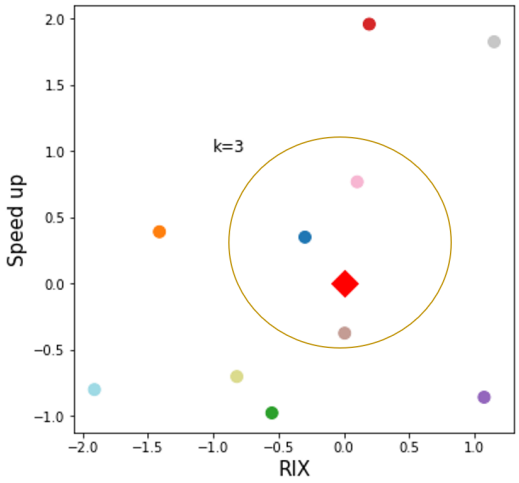

Figure 1. Illustration of the *k*-NN approach, for the case of the features' speedup and RIX. Red diamond: target site. The circles groups the $k = 3$ nearest neighbors out of 10 possible predictors.

2.2. Hyperparameter Estimation in k-NN Models

In a *k*-NN regression, the number of nearest neighbors in feature space k is a *hyperparameter* that determines the accuracy of the model. The other hyperparameter used in this work is the type of weighting w_i (uniform or inverse-distance weighting); see Equation (1). Both hyperparameters are evaluated systematically in a process called *parameter tuning*, which determines the optimal value of k so that prediction errors are minimized. The standard methodology is based on splitting the data set into a training and a testing set, in such a way that the training set allows to estimate the optimal k^* value (usually by cross-validation to avoid over-fitting). Then, the selected k^* value is evaluated on the testing set (unseen data) to obtain an unbiased estimation of the model's performance. In the present work, this approach was used for reference purposes (see below), however, the main contribution of this work to the *k*-NN methodology is the estimation of the optimal hyperparameters without the target location, while still using the full observational period, thereby avoiding seasonality biases; a detailed description can be found in Section 2.4. The two conceptually different approaches used in the work for hyperparameter estimation can then be described as follows:

1. In the first approach, the complete annual set of wind measurements was used to obtain the k^* minimizing the prediction errors for each site. In a first step (termed method kNN0 in Section 2.4), data from all locations, including the target site, were used for that purpose. This step provides a fit to the data, not a prediction. It does, however, allow creating prior knowledge of the best hyperparameters possible for the reference sites. An estimation of the optimal hyperparameters *without* using data from the target site, required for wind resource estimation at a location without measured data, is then conducted in a separate step, as described in Section 2.2.1.
2. The second approach consists of the implementation of a nested cross-validation method, similar to the standard procedure of splitting the data set into a training and evaluation set. As opposed to the standard procedure, however, biases are avoided by using training periods of variable lengths, while maintaining validation and testing periods at constant lengths; as can be seen in Section 2.2.2.

2.2.1. Hyperparameter Estimation with the Full Data Set

The parameter estimation for the full observational period is based on the following four algorithms. Algorithm 1 estimates the wind speed at each time step i using multiple combinations of features and a maximum of k neighbors. The uniformly weighted *k*-NN average $\hat{y}_i$ and the inverse-distance weighted *k*-NN prediction $\hat{y}_i^{(w)}$ were used to estimate the wind speed. The mean percentage error MPE was used as the main error metric.

Algorithm 1 *k*-NN regression with subset and number of neighbors selection

Let X be the matrix of predictors and y be the vector of target values
Let x_0 be the vector of query predictors
Let $\mathcal{F}$ be the set of all possible combination of features
Let K be the maximum number of neighbors to consider
Let $\overline{z} \equiv \frac{1}{N}\sum_{i=1}^{N} z_i$

for each $f_\alpha \subseteq \mathcal{F}$ **do**
  for $k \leftarrow 1$ to K **do**
    for $i \leftarrow 1$ to N **do**
      $D_{i,j} \leftarrow ||x_j - x_{0i}||_2, \forall x_j \in X$ ▷ Variable D stores the L2 norm between query site x_{0i} and each predictor site x_j for each instance i.
      $w_{i,j} \leftarrow \frac{1}{D_{i,j}}$
      Let N_0 be an empty set
      for $i \leftarrow 1$ to k **do**
        $N_{i+1} \leftarrow N_i \cup \arg\min_{x_p | X \setminus N_i} (D)$ ▷ The set N stores the nearest neighbors to the query site for each instance i.
      end for
      $\hat{y}_i \leftarrow \frac{1}{k}\sum_{x_i \in N_k}^{k} y_i$
      $\hat{y}_i^{(w)} \leftarrow \frac{\sum_{x_i \in N_k}^{k} y_i w_i}{\sum_{x_i \in N_k}^{k} w_i}$
    end for
    $MPE_{\alpha,k} \leftarrow \left(\frac{\overline{\hat{y}} - \overline{y}}{\overline{y}}\right) \times 100\%$ ▷ The MPE variable stores the percentage error for each set of features α and number of neighbors k
    $MPE_{\alpha,k}^{(w)} \leftarrow \left(\frac{\overline{\hat{y}^{(w)}} - \overline{y}}{\overline{y}}\right) \times 100\%$
  end for
end for
$MPE_{\alpha,k} \leftarrow MPE_{\alpha,k} \cup MPE_{\alpha,k}^{(w)}$
return $f_\alpha, k, MPE_{\alpha,k}$

Algorithm 2 was used to determine the best hyperparameter k and its associated regression type c, either uniform or inverse-distance weighted. Those best ks were selected for each site and for each set of features based on their MPE values. A ranking of the best set of features for the study site was determined by *cardinality* (i.e., the number of features in a set) based on the average of the MPE of all sites. Here, the four highest-ranking sets of features (i.e., the four best indicators of similarity) are reported in all cases.

Algorithm 2 Selection of the optimal k and best predictors for cardinality 2 to $\#\{X_0\}$

Let X be the matrix of predictors and y be the vector of target values
Let x_0 be the vector of query predictors
Let $\mathcal{F}$ be the set of all possible combination of features
Let K be the maximum number of neighbors to consider
Let *Sites* be the set of sites that will be predicted

for each $site \in Sites$ **do**
  Algorithm 1 $(X, x_0, y, \mathcal{F}, K)$
  $MPE_{site,\alpha,k}$
end for
return $MPE_{site,\alpha,k}$
for each $site \in Sites$ **do**
  for each $f_\alpha \subseteq \mathcal{F}$ **do**
    $k_{site,\alpha}^{(MPE)} \leftarrow \arg\min_k (MPE_{site,\alpha,k})$ ▷ Selection of optimal hyperparameter k^* for each site and set of features indexed by α
  end for
end for
Let $F_0^{(MPE)}$ be an empty set
for $c \leftarrow 2$ to $\#\{X_0\}$ **do**
  $F_{c+1}^{(MPE)} \leftarrow F_c^{(MPE)} \cup \arg\min_{f_\alpha | F \setminus F_c^{(MPE)}, \#\{f_\alpha\}=c} \left(\frac{1}{\#\{sites\}}\sum_{site=1}^{\#\{sites\}} |MPE_{site,\alpha,k_{site,\alpha}^{(MPE)}}|\right)$ ▷ Selection of features f_α that minimize the average MPE of all sites per cardinality
end for
return $F_{\#\{X_0\}}^{(MPE)}$

As mentioned previously, the optimal hyperparameters (k^* and c^*) determined by Algorithm 2 (also referred to as method kNN0, as can be seen in Section 2.4) were determined using the wind measurements of all sites and are therefore not suitable for the estimation of the wind resource at a target location without on-site measurements. To overcome this limitation, the kNNa method (Section 2.4) was designed to estimate the hyperparameters of an arbitrary target site using the optimal hyperparameters of neighboring reference sites, where the estimation is based on the similarity between sites. An illustration of this approach is given in Figure 2. A k-NN classifier was used to predict the two target variables $\hat{k}_{site}$ and $\hat{c}_{site}$ at the target site by the majority or weighted majority class of its k nearest neighbors. This procedure was conducted using Algorithm 3, where an in-depth exploration of the parameters was performed using multiple combinations of features and a number of neighbors in the classifier. Each combination of parameters is called a different classifier. Given that hyperparameters in the present study are not time-dependent, all the features used to estimate $\hat{k}$ and $\hat{c}$ must be constant in time. Therefore, a mean value was calculated for time-dependent variables.

Algorithm 3 k-NN classification with subset and number of neighbors selection

Let $\overline{\overline{X}}$ be the matrix of mean predictor values for all sites
Let k^* and c^* be the vector of optimal parameters for k and c for all sites
Let $\overline{x}_0$ be the vector of mean predictor values at query location
Let $\mathcal{F}$ be the set of all possible combinations of features
Let $\#\{Sites\} - 1$ be the maximum number of neighboring sites

for each $f_\alpha \subseteq \mathcal{F}$ **do**
  for $j \leftarrow 1$ to $\#\{Sites\} - 1$ **do**
    for each $site \in Sites$ **do**
      $D_{site,q} \leftarrow ||x_q - \overline{x}_0||_2, \forall x_q \in \overline{\overline{X}} | x_q \neq \overline{x}_0$
      $I_{site,q} \leftarrow \frac{1}{D_{site,q}}$
      $W_{site} \leftarrow \sum_{q=1}^{j} I_{site,q}$
      $w_{site,q} \leftarrow \frac{I_{site,q}}{W_{site}}$
      Let Q_0 be an empty set
      for $i \leftarrow 1$ to j **do**
        $Q_{i+1} \leftarrow Q_i \cup \arg\min_{q|Sites \setminus Q_i} (D_{site,q})$
      end for
      Let k_γ^* be the set of unique k_l^* denoted as $\{k_1^*, ..., k_L^*\}_{\neq}$ ▷ The subset of unique values of k^* is given by k_γ^*
      $vote_{k_l^*} \leftarrow \sum_{q \in Q_j} I(k_q^* = k_l^*), \forall l \in k_\gamma^*$ ▷ $vote_{k_l^*}$ saves the votes for each class k_l^* from neighborhood Q_j
      $\hat{k}_{\alpha,j,site} \leftarrow \arg\max_{k_l^*}(vote_{k_l^*})$ ▷ The majority class is the predicted number of neighbors $\hat{k}$, estimated with the classifier f_α and number of neighbors j
      $vote_{k_l^*}^{(w)} \leftarrow \sum_{q \in Q_j} I(k_q^* = k_l^*) w_{site,q}, \forall l \in k_\gamma^*$
      $\hat{k}_{\alpha,j,site}^{(w)} \leftarrow \arg\max_{k_l^*}(vote_{k_l^*}^{(w)})$
      Let c^* be a set of two labels $\{c_1, c_2\} = \{uniform, distance\}$
      $vote_{c_l^*} \leftarrow \sum_{q \in Q_j} I(c_q^* = c_l^*), \forall l \in c_l^*$ ▷ $vote_{c_l^*}$ saves the votes for each class c_l^* from neighborhood Q_j
      $\hat{c}_{\alpha,j,site} \leftarrow \arg\max_{c_l^*}(vote_{c_l^*})$ ▷ The majority class is the predicted type of regression $\hat{c}$, estimated with the classifier f_α and number of neighbors j
      $vote_{c_l^*}^{(w)} \leftarrow \sum_{q \in Q_j} I(c_q^* = c_l^*) w_{site,q}, \forall l \in c_l^*$
      $\hat{c}_{\alpha,j,site}^{(w)} \leftarrow \arg\max_{c_l^*}(vote_{c_l^*}^{(w)})$
    end for
  end for
end for
$\hat{k}_{\alpha,j,site} \leftarrow \hat{k}_{\alpha,j,site} \cup \hat{k}_{\alpha,j,site}^{(w)}$
$\hat{c}_{\alpha,j,site} \leftarrow \hat{c}_{\alpha,j,site} \cup \hat{c}_{\alpha,j,site}^{(w)}$
return $\hat{k}_{\alpha,j,site}, \hat{c}_{\alpha,j,site}$

To determine which of the k-NN classifiers evaluated in Algorithm 3 had the best performance, wind speed predictions were conducted using the predicted hyperparameters $\hat{k}$ and $\hat{c}$. The classifier minimizing the average error of the reference sites (leaving out the target site) was selected. Each site at its turn was assumed to be a target site, therefore N_{site}

(number of sites) average errors were calculated; the classifier that repeated the most in those performance rankings was then selected. A pseudocode description of the method described is shown in Algorithm 4.

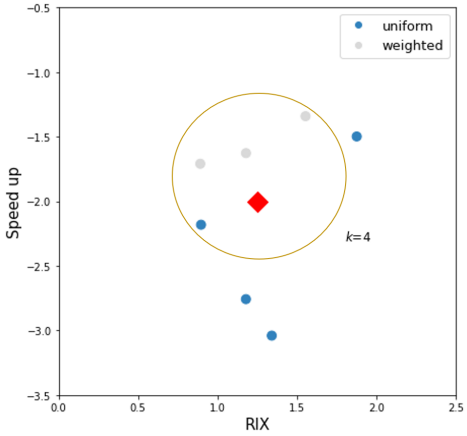

Figure 2. Illustration of the k-NN classifier method to predict the type of regression $\hat{c}$ (uniform or inverse-distance weighting), used by Algorithm 3. Red diamond: target site. The circles group the $k = 4$ nearest neighbors in the 2-dimensional space created by the features *speedup* and RIX. The same idea holds for predicting the number of neighbors $\hat{k}$.

Algorithm 4 k-NN regression to estimate $\hat{y}$ using estimated hyperparameters

Let X be the matrix of predictors and y be the vector of target values
Let x_0 be the vector of query predictors
Let $\mathcal{F}$ be the set of all possible combinations of features
Let $F^{(MPE)}_{\#\{X_0\}}$ be the set of best predictors

for each $f_\beta \subseteq F^{(MPE)}_{\#\{X_0\}}$ **do** ▷ The k-NN regression uses the set of selected features ($F^{(MPE)}_{\#\{X_0\}}$) by Algorithm 2, and indexed by β
  for each $site \in Sites$ **do**
    for each $\hat{k}_r \in \hat{k}_{\alpha,k,site}$, $\hat{c}_r \in \hat{c}_{\alpha,k,site}$ **do** ▷ The hyperparameters $\hat{k}$ and $\hat{c}$ predicted by the k-NN classifiers are indexed by r
      for $i \leftarrow 1$ to N **do**
        $D_{i,j} \leftarrow ||x_j - x_{0i}||_2, \forall x_j \in X | x_j \neq x_0$
        $w_{i,j} \leftarrow \frac{1}{D_{i,j}}$
        Let J_0 be an empty set
        for $i \leftarrow 1$ to $\hat{k}_r$ **do**
          $J_{i+1} \leftarrow J_i \cup \arg\min_{x_p | X \setminus J_i} (D)$
        end for
        if $\hat{c}_r = uniform$ **then**
          $\hat{y}_{i,\beta,site,\hat{k}_r} \leftarrow \frac{1}{\hat{k}_r} \sum_{x_i \in J_{\hat{k}_r}}^{\hat{k}_r} y_i$
        else
          $\hat{y}_{i,\beta,site,\hat{k}_r} \leftarrow \frac{\sum_{x_i \in J_{\hat{k}_r}}^{\hat{k}_r} y_i w_i}{\sum_{x_i \in J_{\hat{k}_r}}^{\hat{k}_r} w_i}$
        end if
      end for
      $MPE_{\beta,site,\hat{k}_r} \leftarrow \left(\frac{\overline{\hat{y}}_{\beta,site,\hat{k}_r} - \overline{y}_{site}}{\overline{y}_{site}} \right) \times 100\%$ ▷ The $MPE_{\beta,site,\hat{k}_r}$ variable stores the errors using features β, at a given $site$ with the estimated $\hat{k}$ and $\hat{c}$
    end for
    $r^{(MPE)}_{\beta,site} \leftarrow \arg\min_r \left(\frac{1}{\#\{Sites\} - 1} \sum_{\substack{i=0 \\ i \neq site}}^{\#\{Sites\}-1} |MPE_{\beta,site,\hat{k}_r}| \right)$ ▷ Selection of the classifier that belongs to the index r that minimizes the avg. error leaving one site out
  end for
  Let η_α be the set of unique $r^{(MPE)}_{\beta,l}$ denoted as $\{r^{(MPE)}_{\beta,1}, ..., r^{(MPE)}_{\beta,L}\}_{\neq}$
  $vote_{r^{(MPE)}_{\beta,l}} \leftarrow \sum_{site \in Sites} I(r^{(MPE)}_{\beta,site} = r^{(MPE)}_{\beta,l}), \forall l \in \eta_\alpha$ ▷ $vote_{r^{(MPE)}_{\beta,l}}$ saves the votes for each classifier
  $r^{(MPE)}_{\beta,opt} \leftarrow \arg\max_{r^{(MPE)}_{\beta,l}} (vote_{r^{(MPE)}_{\beta,l}})$ ▷ $r^{(MPE)}_{\beta,opt}$ saves the classifier that repeat the most for each set of features β
end for
return $\overline{\hat{y}}_{\beta,site,r^{(MPE)}_{\beta,opt}}$

2.2.2. Hyperparameter Selection and Testing through Nested Cross-Validation

As described above, often *k*-NN methods deals with time series data, which portray different characteristics when analyzed within different periods. For hyperparameter tuning and providing an unbiased validation of a given method for modeling or predicting a variable (such as wind speed), cross-validation is commonly carried out. In the present work, the nested cross-validation approach illustrated in Figure 3 was implemented. With such an approach, the testing period (green square in Figure 3) appears chronologically after the training period (blue squares). The training data are used to systematically evaluate the number of neighbors and type of regression, and the setup that minimizes the MPE is selected and used in the testing set to provide an unbiased measure of the model prediction error. By repeating this process with multiple testing sets, a better estimate of model performance is obtained. In this work, the data set was split into five nested subsets, using the setup shown in Table 1. The testing period was constant in all runs (2 months). As mentioned before, this implementation was conducted for reference purposes only.

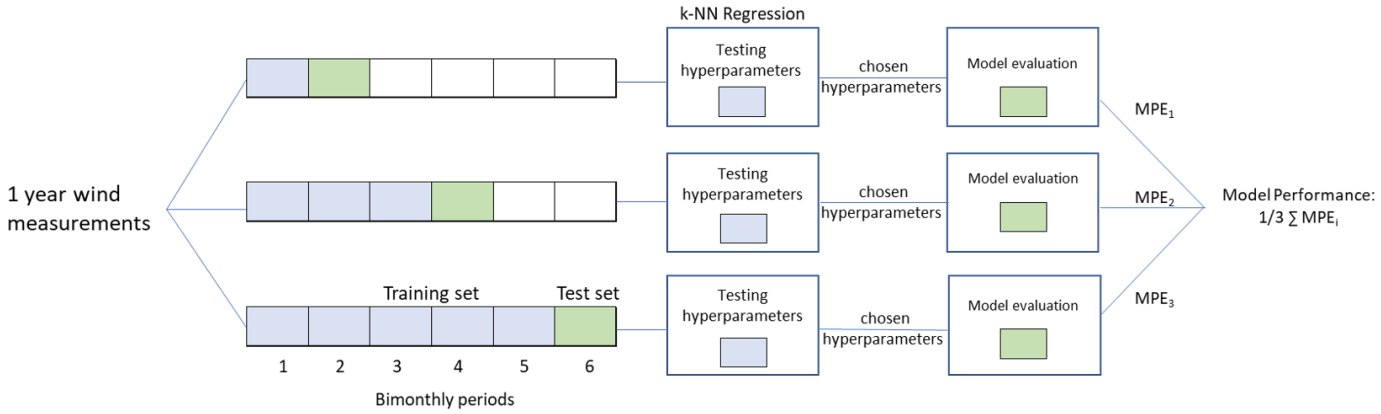

Figure 3. Illustration of the nested hyperparameter estimation and cross-validation approach.

Table 1. Setup of nested cross-validation simulations.

Data set/case	1	2	3	4	5
Total observations	17,568	26,352	35,136	43,920	52,704
Training observations	8784	17,568	26,352	35,136	43,920
Testing observations	8784	8784	8784	8784	8784

2.3. Feature Selection for k-NN Methods

A number of features were considered for their use with *k*-NN-based methods. A basic requirement for all candidates was the possibility of calculating their values for each location of interest from geographic information alone, in a completely analogous way in which flow modeling tools such as WAsP or WindSim construct wind maps for a site or region of interest. The following feature candidates were found to be suitable candidates:

- **The terrain ruggedness index RIX**. Although the ΔRIX correction procedure [18] did not provide a significant improvement of the WAsP flow modeling results for the site, a weak correlation between the WAsP flow modeling error for the wind speed indicated a possible suitability of the RIX index as a similarity feature.
- **The orography-induced speedup** (e.g., relative increase in wind speed due the terrain slope). A modest correlation between the prediction error and the difference between the WAsP-calculated topography-induced speedups was found (Figure 4), indicating that the speedup is a possible similarity measure candidate.
- **The distance between measurement locations** (met towers). As shown in Figure 4, the distance between locations bears a similar impact on the flow modeling results as the difference between speedup values.

- **The vertical wind shear**. The wind shear was found to have a significant (negative) correlation with the prediction error, and was therefore expected to be among the important predictor variables of the *k*-NN method. Note that the degree of correlation between the flow modeling error and the feature candidates shown in Figure 4 does not have any impact on the *k*-NN methodology, beyond the decision of including or not a given candidate feature in the list of *k*-NN predictors. The *k*-NN method determines the optimal set of predictors in an automated fashion, as described above.

The following additional feature parameters were used for the assessment of power density:

- **The Weibull scale and shape parameters**. Determining Weibull parameters requires on-site measurements (or simulated winds from atmospheric models) for at last one location. Using such a reference location, a *transferred* wind rose (the latter understood as the set of angular wind speed histograms) can be constructed using a wind flow modeling tool. Since local flow conditions are different for different wind directions, the average transferred wind rose will generally have different Weibull parameters.
- **The R^2-value, or coefficient of determination, of the Weibull fit**. The transfer of wind roses may not only change the Weibull parameters, but may also distort the underlying histogram, resulting in varying degrees of goodness of fit. R^2-related parameters can be constructed for their use as similarity features; additionally, a modified weighting scheme (similarly to the one proposed in [19]) was explored as well, where the R^2-value was used to build a confidence level matrix as part of a generalized inverse-distance averaging scheme.

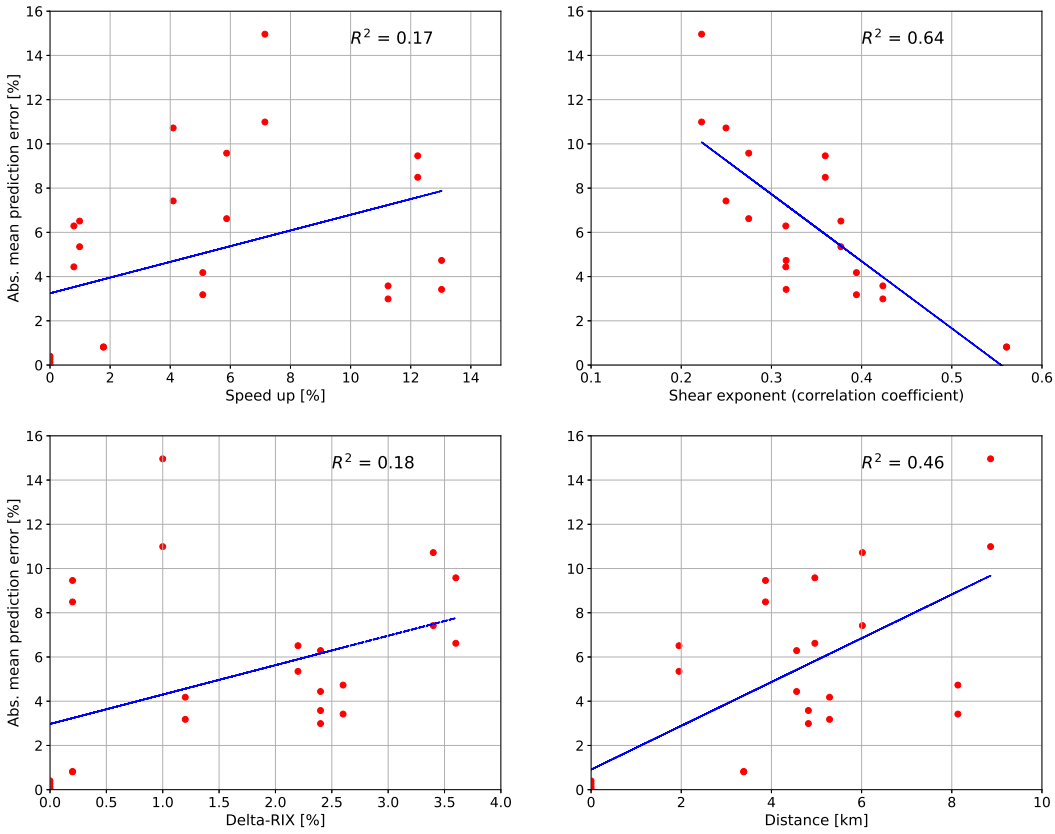

Figure 4. Mean absolute modeling error for WAsP-predicted wind speed vs. features selected for the *k*-NN study. The dots correspond to the cross-validation between each pair of sites. The blue line represents the linear regression of the errors as a function of presented features.

2.4. Setup of the k-NN Simulations

The current work consists of the following sequence of assessments:

1. **KNN0**. All available wind speed data for the full one-year period (see Section 2.5) were used to perform a *k*-NN regression for each of the five towers as target site. The optimal hyperparameters k^* and c^* were calculated from the full regression. Algorithms 1 and 2 were used for this purpose. This is a baseline case, constructed for reference purposes.
2. **KNNa**. In order to conduct an independent test of the methodology, the optimal hyperparameters were estimated from all towers, excluding the target tower itself. The corresponding procedure is described in Algorithm 3.
3. **KNNb**. Instead of using the measured wind speed information directly as a predictor for a given target site, an alternative method consists of using the WAsP predictions for the target site, prepared with each of the predictor sites. The same set of features and feature combinations as before were used in this step.
4. **KNNc**. Since methods KNNa (driven exclusively with observed wind speed data) and KNNb (working on WAsP-processed observational data) represent independent assessments, an *ensemble* version was performed, where methods KNNa and KNNb were combined linearly.
5. **KNNd**. For an independent validation of the *k*-NN approach, an additional method was implemented, where optimal hyperparameters were determined and validated in the nested approach described in Section 2.2.2.

2.5. Validation Data

On-site tall tower meteorological data from the development phase of a commercial wind farm in Mexico were used for model construction and validation. The data are proprietary and are not in the public domain. Each of the five towers at the site ("site B") was equipped with three pairs of redundant cup anemometers (class I for primary sensors, standard for redundant) placed at 80, 60, and 40 m above ground level. The wind direction was measured at two levels (42 and 78 m). Temperature measurements were taken at 12 and 80 m above ground level. Data were recorded at 10 min intervals; for each variable, the mean, maximum, minimum and standard deviation were recorded. One full year of concurrent information with only minor data gaps was selected to avoid seasonal biases. Initial quality assurance was conducted in a semi-automatic way using Windographer. Overall data recovery after quality assurance was 99.9%. All reported results for the wind speed modeling accuracy in this study refer to the 80 m wind speed.

In order to build a continuous observational period, three reconstruction methods were used: (1) replacement of missing or invalid 80 m data with those from the redundant sensor, prior tower shadow correction when necessary; (2) vertical extrapolation, and (3) principal component analysis (PCA). Vertical extrapolation was used when 40 m and 60 m wind speed data were available. PCA was used to take advantage of concurrent data from other towers where no concurrent wind speed data at the same tower were available.

In order to ascertain that the reconstructed continuous data records for each met tower were statistically indistinguishable from the quality-controlled original data, both a χ^2 test for the observed ($f_O(x_i)$) and reconstructed ($f_S(x_i)$) wind speed distributions, with $\chi^2 = \sum(f_O(x_i) - f_S(x_i)/f_O(x_i)$, and a Kolmogorov–Smirnov test for the cumulative probability density functions ($F_O(x_i)$ and $F_S(x_i)$, respectively) were conducted. All tower measurements at site B successfully passed both tests, demonstrating that the quality-controlled original data sets and the reconstructed full data sets were statistically indistinguishable.

The wind regime at the site is essentially bi-modal, with Southerly winds predominating most of the year, the remainder corresponding to Northerly winds (see Figure 5). All towers were located at either the Southern or the Northern edge of a plateau structure, with three towers (S01_B, S02_B, SO3_B) being located at the Southern and two (S04_B, S05_B) at the Northern edge. S01_B and S05_B are somewhat more exposed locations, sitting on extruded portions of the plateau, whereas the other towers sit between a steep slope and the flat upper part of the plateau.

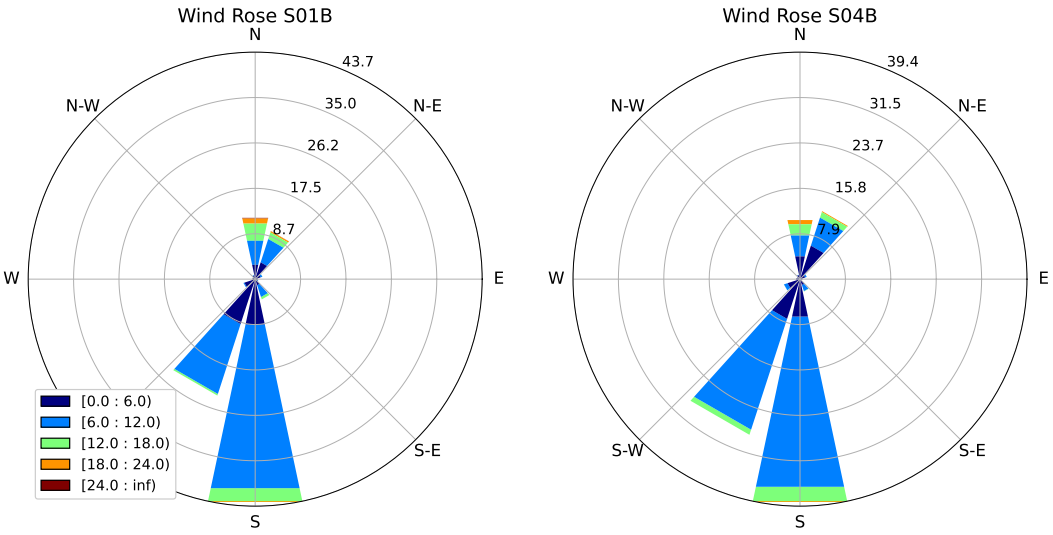

Figure 5. Wind rose observed at sites S01B and S04B from June 2016 to June 2017.

2.6. Flow Modeling: WAsP

WAsP [5] was used for (1) wind flow modeling and cross-prediction of the wind resource at all locations with observational data; and (2) the determination of the feature parameters (Section 2.4). Orography inputs come from the Shuttle Radar Topography Mission (SRTM) digital elevation data [20] with a resolution of 1 arcsec (30 m). The SRTM 1 Arc-Second Global version [21] used was downloaded from the U.S. Geological Survey (USGS) website. The domain extends 10 km beyond the limits of the site of interest in order to create a large enough buffer to avoid boundary effects; contour lines with a spacing of 10 m were used. The GlobeLand30 (GLC30) database [22] with a resolution of 1 arcsec (30 m) was used to create a roughness map. Using the GLC30 version 2020 database improves the cross-validation results for the wind speed compared to using the WAsP default database (GlobCover-2009 map [23]); therefore, we use the former roughness data hereafter. A detailed discussion of the impact of potentially more accurate roughness maps and its impact of flow modeling accuracy is not part of the present work.

In the WAsP modeling chain, the observed wind is generalized using predefined heights and roughness lengths, which can be modified. It was observed that a fine tune of these values reduces the error in the predicted wind. Here, the predefined heights were set to 10, 30, 60, 80 and 100 m above ground level and the standard roughness lengths were set to 0.0, 0.05, 0.11, 0.23 and 0.5 m.

In order to account for atmospheric stability, a systematic assessment of the heat flux impact on the vertical wind shear profiles was conducted (not shown). By using the WAsP default configuration, corresponding to a slightly stable condition with a heat flux offset of -40 W/m^2, a good fit with the observed profiles was found at all tower locations. Allowing for heat flux offset variations did not improve the results. Another option in the WAsP software is the geostrophic shear model, which is turned on by default. The geostrophic shear is obtained from coarse reanalysis data, which cannot resolve the slopes in mountainous terrain [24], and can therefore give unphysically large geostrophic wind shear values. This was also observed for the study site in this work (0.02 m/s/m). The model was therefore turned off in all cases.

An attempt to further improve the results of the cross-validation predictions using the delta-RIX methodology [18] was performed. However, as mentioned in Section 2.3, no statistically significant improvement was obtained.

As an additional reference, the WAsP software was run in (RANS) CFD mode; the results were found to be similar to the ones obtained with WindSim, discussed in the next subsection. Unless mentioned otherwise, all WAsP results were obtained with the standard (linear) flow solver.

2.7. Flow Modeling: WindSim

A RANS-based solver, the commercial software package WindSim ([25]), was used in an attempt to improve cross-predictions, given the terrain complexity of the site. The same topography data (STRM and GLC30) as those in WAsP were used. The WindSim mesh was setup with an initial horizontal resolution of 50 m in the refinement area for convergence and initial performance tests; the resolution in the refinement area used in the final selected configuration was then set to 20 m for a better representation of the terrain at its steep parts. Setup parameters for WindSim include (1) a toggle parameter for including the forest model (on/off); (2) the free parameters of the forest model; (3) the turbulence closure model (k-ϵ or k-ω); and (4) the type of atmospheric stability. A Taguchi experiment design [26] was set up in order to assess the different free parameters of the forest model. The assumption of neutral stability was found to produce a good agreement with the observed wind speed profiles; moreover, simulations under stable and unstable conditions lead to convergence problems, so only neutral stability conditions were considered for the final model setup. The use of the k-ω turbulence closure model and the omission of the forest model produce the best results generally, so we use this configuration hereafter. A detailed discussion of the setup, experiment design and results obtained with the WindSim study are beyond the scope of the present work and will be reported elsewhere.

3. Results

3.1. Cross-Validation with Flow Models

As baseline, we choose the best results from WAsP and WindSim, which are summarized in Figure 6, where the cross-validation errors (in %) for the prediction of the average wind speed obtained with WAsP (a) and WindSim (b) are shown. In both cases, a relatively small prediction error is observed within groups S (S01B, S02B, S03B), which are locations at the Southern edge of the plateau, and N (S04B, S05B), which are at the Northern edge. The error is dramatically larger when cross-predicting between these two groups. WindSim does not seem to systematically reduce the latter errors.

(a) WAsP

Predicted site \ Predictor site	S01B	S02B	S03B	S04B	S05B
S01B	0.11	3.61	4.71	9.50	14.96
S02B	-2.95	0.00	0.81	6.48	10.67
S03B	-3.34	-0.85	-0.22	6.25	9.54
S04B	-8.45	-5.30	-4.40	0.01	4.24
S05B	-11.00	-7.37	-6.62	-3.24	-0.35

(b) WindSim

Predicted site \ Predictor site	S01B	S02B	S03B	S04B	S05B
S01B	0.00	0.39	4.50	8.17	10.50
S02B	-0.03	0.00	4.00	8.61	10.38
S03B	-3.54	-3.84	0.00	4.93	5.50
S04B	-6.91	-6.43	-2.79	0.00	1.63
S05B	-8.32	-8.06	-4.12	-1.55	0.00

Figure 6. Cross−validation errors for the prediction of the average wind speed in % obtained with WAsP (**a**) and WindSim (**b**). Light orange colors: small errors; reddish colors: large errors.

Figure 7 shows the cross-validation results as a function of the distance between predictor-predicted pairs. This distance highly influences the prediction accuracy, but only within the northern (N) and southern (S) zones. The absolute prediction errors are similar within the zones. However, inter-zone cross-predictions show a much larger error than intra-zone cross-predictions. This is as expected, as the met towers are located near either of the plateau edges. WindSim demonstrated good results in the Bolund benchmark [11], so one might expected improvements compared to WAsP at this site. WindSim results are

only slightly better than WAsP and both models show (≈10%) errors when cross-predicting between zones.

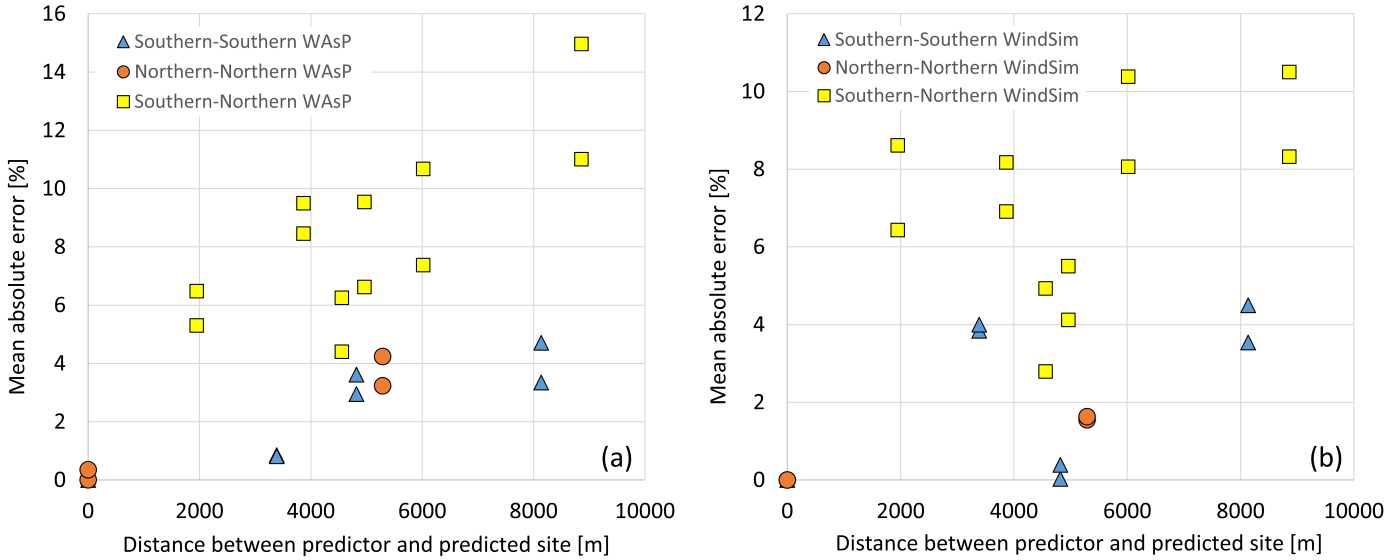

Figure 7. Absolute mean prediction error of the cross-validations as a function of distance and zone for the results using (**a**) WAsP; and (**b**) WindSim.

Based on the above findings, which indicate that the cross-prediction accuracy between zones is limited, we can assume that the wind climatology at the S and N edges of the plateau is significantly influenced by the local orography. Using an inverse-distance weight approach does not improve cross-predictions as shown in Table 2 (column 3) for the case of WAsP. The lack of accuracy continues to be particularly noticeable at the edges of the plateau (locations S01B and S05B), where the wind speed errors are of the order of 7–8%, which are much larger than the accepted error for wind farm development.

Table 2. Percentage error of the cross-predictions of four implementations of the *k*-NNa method (kNNa card. 2 − 5) operating for cardinalities 2, 3, 4 and 5, compared to the flow modeling results using WAsP. WAsP = average of all WAsP cross-predictions for a given target met tower site; IDW = inverse-distance averaged WAsP cross-predictions.

Tower	WAsP (Linear)	WAsP (Linear, IDW)	kNNa Implementations (Card.2)	(Card.3)	(Card.4)	(Card.5)
S01_B	8.19	7.80	−3.39	−3.49	−3.91	−3.95
S02_B	3.75	4.00	−0.85	−0.60	0.03	0.14
S03_B	2.90	3.13	1.55	1.94	0.96	0.96
S04_B	−3.48	−4.30	−1.57	−0.22	−0.99	−3.28
S05_B	−7.06	−6.59	0.49	0.86	−0.54	−0.36
MAE (%)	5.08	5.16	1.57	1.42	1.29	1.74

3.2. k-NN Modeling Results

3.2.1. Overall Results

As described in Section 2.4, the kNNa model uses estimated, rather than fitted, hyper-parameters, and therefore provides a true prediction of the wind speed at each location, as opposed to a fit to the data. Input data are the measured wind speeds at each of the towers, excluding those of the target location. As shown in Table 2, where the results for four implementations (see Section 2.2.1) of the kNNa method are shown, alongside the results from WAsP, the overall performance of the *k*-NN method is far better than that of the flow model. Whereas the uniformly (column 2) and inverse-distance weighted (column 3)

average WAsP results have an error of about 5%, the *k*-NN error is about 1.5%, with the best-ranking method (kNNa.card4, using cardinality 4, see Table 3) yielding an error of only 1.29%.

It is also worth noting that not only is the average error of the *k*-NN predictions significantly smaller than the one obtained with WAsP, but the discrepancies at the more exposed locations are also largely resolved. While the flow model produces deviations of about 8 and −7% for S01B and S05Bs, respectively, the corresponding *k*-NN results are around 3.75 and 0.5%, respectively.

3.2.2. Hyperparameter Optimization

In order to build some intuition of the structure of the results of the *k*-NN method, the KNNa case described above will be further discussed. As described above, KNNa uses estimated hyperparameters ($\hat{k}$ and $\hat{c}$) for each target site, rather than their optimal values (as determined by method KNN0). The modeled wind speeds are therefore *independent* predictions, not the result of a regression. The results are shown in Table 3 for each target site and each level of cardinality. The method with cardinality 4 (based on features' distance, RIX, shear exponent, and time) produces the smallest error (1.29% average), though good results are obtained with the other methods as well. In all cases, the error is much smaller than in the case of the flow modeling predictions (WAsP or WindSim). It can also be seen that the number of nearest neighbors ($\hat{k}$) used for the prediction of the wind speed at each location varies among methods. Whereas the cardinality-2 method requires up to nine nearest neighbors and shows great variability in the hyperparameter $\hat{k}$ among sites, the results for the cardinality-4 method appear to be more consistent among the tower locations for both $\hat{k}$ and $\hat{c}$, with the best predictions for each location based on one or two adjacent locations and using uniform weighting in all cases. To understand why a location can have nine neighbors in a five-tower arrangement, it should be recalled that *time* is used as a feature parameter as well, allowing the wind speed at time step t_n to be modeled based on wind speeds at times t_{n-1} and t_{n+1}. It should also be stressed that the aim is to predict wind speeds and not to forecast them (in time), which would preclude future time steps to be considered. In any case, restricting predictor wind speeds to simultaneous time steps still only produces excellent results, as evidenced by the cardinality-3 and -4 methods, with method 3 being the highest-ranking method of all.

Table 3. Highest-ranking *k*-NN solutions for case kNNa and each cardinality (number of features considered). Estimated optimal hyperparameters $\hat{k}$ and $\hat{c}$ are shown, along with their feature sets and their mean percentage error for each target site.

Cardinality	Parameter	Features	S01B	S02B	S03B	S04B	S05B	MAPE (%)
2	MPE (%)	RIX and time	−3.39	−0.85	1.55	−1.57	0.49	1.57
	$\hat{k}$		2	1	9	9	9	
	$\hat{c}$		uniform	uniform	distance	uniform	uniform	
3	MPE (%)	RIX, shear exponent and time	−3.49	−0.60	1.94	−0.22	0.86	1.42
	$\hat{k}$		9	1	2	2	9	
	$\hat{c}$		distance	uniform	distance	uniform	distance	
4	MPE (%)	Distance, RIX, shear exponent and time	−3.91	0.03	0.96	−0.99	−0.54	1.29
	$\hat{k}$		2	1	1	2	2	
	$\hat{c}$		uniform	uniform	uniform	uniform	uniform	
5	MPE (%)	Distance, RIX, shear exponent, speedup and time	−3.95	0.14	0.96	−3.28	−0.36	1.74
	$\hat{k}$		3	1	1	3	4	
	$\hat{c}$		distance	uniform	uniform	distance	uniform	

3.2.3. Results Obtained the Hybrid WAsP-KNN Approach

Here, we first predict the wind speed at a location (e.g., by using WAsP) and then we use these predictions at the target location from each predictor site within the *k*-NN model framework. In other words, this implementation works with the *transferred climatologies*, rather than with the measured wind speed time series.

Results obtained with this method (referred to as KNNb in Section 2.4) are shown in Table 4 under the column tagged "hybrid". The method does not result in a decrease in the error at the more exposed locations (S01B and S05B), although a slight improvement at the less exposed locations occurs. The MAPE is comparable to that of WindSim.

3.2.4. Results Obtained with the Ensemble Approach

The ensemble approach (kNNc) consists of a linear superposition of models kNNa and kNNb, where kNNa is driven by the observed wind speed data at each location, and kNNb works on the transferred microclimates from each predictor tower to the target location.

In order to determine the statistical weight of each method, first a regression was conducted, leading to optimal weights for each target site. In a second step, average weights were used to build a new prediction. The accuracy observed was somewhat better on the average (around 1.1%) than the observational data-based *k*-NN results, and showed a better consistency among target sites. Since the two methods used (kNNa and kNNb) are independent predictions of the wind resource, as their combination can lead to errors being canceled out—as shown in Table 4, where the average error is about 1%, with a quite homogeneous prediction accuracy at all locations.

3.2.5. Results Obtained with the Nested Approach

The nested approach (kNNd) cannot be directly used to estimate the wind resource at a location without on-site measurements (as opposed to methods kNNa, kNNb, and kNNc) and was implemented for reference purposes only. Given that kNNd uses the full set of data, the metrics of kNNd are very good, with an average error of about 1%, comparable to the best performing methods kNNa and kNNb. Note that kNNd does a better job at locations (S02B, S03B, SOB) compared to the more exposed locations (S01B and S05B), similarly to kNNa and kNNb, highlighting that the capabilities of a method based on similarity are limited by the number of locations available with similar features. In modern wind resource campaigns, which now often include remote sensing devices such as sodar [27] or lidar [28], this limitation can, however, readily be overcome with complementary remote sensing campaigns.

Table 4. Summary of all wind speed modeling results in this work.

	Flow Models			***k*-NN Methods**			
Tower	**WAsP (linear)**	**WAsP (CFD)**	**Wind Sim**	**Obs.data (kNNa)**	**Hybrid (kNNb)**	**Ensemble (kNNc)**	**Nested (kNNd)**
S01_B	8.19	2.93	5.89	−3.91	8.06	−0.80	−2.15
S02_B	3.75	4.35	5.74	0.03	0.88	0.25	0.26
S03_B	2.90	3.65	0.76	0.96	-0.33	0.62	0.68
S04_B	−3.48	−4.70	−3.63	−0.99	−5.13	−2.07	−0.69
S05_B	−7.06	−8.61	−5.51	−0.54	−4.44	−1.56	1.87
MAPE	5.08	4.85	4.31	1.29	3.77	1.06	1.13

3.2.6. Power Density Results

As described above, estimating power density is potentially more challenging in a complex terrain than estimating wind speed because considerable differences may arise in the simulated wind speed distributions due to terrain effects. As shown in Table 5, the

WAsP predictions for power density had similar errors at four locations and a large error (27%) at one location (S01B). At this location, the wind speed prediction of both WAsP and *k*-NN methods (see Table 5) had also a large error. As this more exposed location, the simulated wind speed distribution differs the most from the observations.

As shown in Table 5, the *k*-NN method (optimized for wind speed prediction) does a considerably better job of predicting power density at SO1B than WAsP (albeit at the expense of worse predictions for S05B), resulting in similar overall errors for the set of five towers as in the case of WAsP. However, by including the location-specific Weibull parameters as additional similarity features, the *k*-NN method can be tuned to provide significantly better power density predictions, as shown in columns 7–10 of Table 5. Prediction errors are now down by approximately 10% for location S01B and on average by only approximately 3 to 3.5% for all sites, which is only half of the error obtained with WAsP.

Table 5. Average percentage error for the prediction of power density obtained with WAsP (linear solver), the original implementation of the *k*-NN method, tuned for wind speed prediction, and a modified *k*-NN version, optimized for the prediction of the power density. The four highest ranking implementations are shown for both *k*-NN cases.

		k-NN (Original Method)				**k-NN (Opt. for Power Dens.)**			
Tower	**WAsP**	**(1)**	**(2)**	**(3)**	**(4)**	**(1)**	**(2)**	**(3)**	**(4)**
S01_B	27.38	−18.29	−16.72	−13.52	−17.82	−10.32	−10.72	−10.21	−9.62
S02_B	1.77	−1.10	−0.88	0.02	0.23	1.02	0.12	0.14	−1.14
S03_B	−3.14	−0.82	0.21	0.74	0.74	−1.90	−0.12	−0.78	0.13
S04_B	1.55	−3.08	−1.90	6.21	−5.23	0.76	1.63	0.83	1.62
S05_B	−1.28	12.19	13.19	8.22	11.42	2.15	3.87	4.72	5.57
MAPE	7.02	7.10	6.58	5.74	7.09	3.23	3.29	3.33	3.62

4. Summary and Conclusions

Here, a novel method for the modeling of the wind resource with improved accuracy was proposed and evaluated using on-site data from a site with complex terrain characteristics. As opposed to industry-standard approaches, which mainly rely on flow models, the new method uses a simple and intuitive machine learning algorithm, based on the *k*-nearest neighbors concept. While being a consolidated method in the literature on advanced statistical modeling and machine learning, to the best knowledge of the authors, no such approach was previously proposed or demonstrated in the field of wind resource assessment.

The *k*-NN method in this work was based on the concept of *similarity* between met tower locations. To assess similarity, the method uses classifiers or *features* of each location. Features related to the terrain characteristics, flow-related quantities and micro-climate parameters were selected. All location features can be readily obtained with a microscale flow model. All *k*-NN runs performed in this work use features generated with WAsP. In order to generate a baseline for comparisons, cross-validations between the five tall (80 m) tower locations were conducted with both WAsP (using its linear flow solver) and WindSim (using a RANS implementation). Both software suites were fine-tuned in order to strive for optimal performance.

Given that all met towers are located at edge locations, a clear improvement of the RANS-CFD predictions over the linear flow model was expected. Some improvement by the best-tuned RANS model was indeed observed. Both flow models failed, however, to accurately transfer the measured climatologies from the northern edge to the southern edge locations and vice versa with the accuracy required by modern competitive wind farms.

Four conceptually different *k*-NN approaches were proposed and assessed, with one of the methods (kNNd) being used for reference purposes. The baseline model (kNNa)

works directly with the quality-controlled wind speed or power density 10 min time series retrieved from all met towers other than the target location used for validation. The hyperparameters required by the *k*-NN model were estimated with a novel methodology based on the similarity between locations, which takes advantage of the full observational period. This is somewhat different from a typical machine-learning approach (implemented as method kNNd) and more suitable for the context of this work.

Two other variants of the *k*-NN concept were implemented as well. In kNNb, the observed wind data were first transferred to the target site using WAsP, and the *k*-NN method was then applied to those transferred climatologies. This hybrid approach has the advantage of being an independent assessment of the wind resource at the target site, allowing for the construction of an *ensemble* method (kNNc) by combining the predictions of kNNa and kNNb.

The *k*-NN concept, in its different implementations, provided significant improvements overflow modeling results. While the average WAsP prediction accuracy was around 5% for the wind speed and 7% for the wind power density, the corresponding figures for *k*-NN method were only approximately 1.5 and 3%, respectively. It should be noted that from the perspective of a wind resource modeler, the *k*-NN method works similarly to a flow model such WAsP or WindSim; only terrain-related information and wind time series for different measurement locations are needed.

While providing improved accuracy over wind flow models, the *k*-NN approach does have some limitations. First, a flow model is needed to determine the *feature* parameters of each location of interest. Therefore, the *k*-NN method can be viewed as an add-on to existing wind modeling suites such as WAsP and WindSim, rather than a stand-alone method. However, wind resource modelers typically have access to flow modeling tools, so this is not a practical limitation. Second, the *k*-NN method needs more than one measurement location, as opposed to WAsP or WindSim. This is, however, not a strong limitation, since nowadays, a number of met towers are routinely installed at prospective wind farm sites, and additional measurement locations can be created dynamically using remote sensing devices such as lidar and sodar. The proposed method was therefore believed to have good prospects for applications in a practical wind resource assessment context.

Author Contributions: Conceptualization, J.L.P. and O.P.; data curation, P.Q.-N. and G.C.; formal analysis, P.Q.-N., J.L.P. and O.P.; funding acquisition, A.P. and O.P.; investigation, P.Q.-N. and G.C.; methodology, P.Q.-N., J.L.P. and O.P.; project administration, A.P. and O.P.; resources, A.P. and O.P.; software, R.F. and G.C.-F.; supervision, J.L.P. and O.P.; validation, R.F.; visualization, P.Q.-N.; writing—original draft, O.P.; writing—review and editing, P.Q.-N., R.F., A.P., G.C.-F. and O.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Danish International Development Agency under grant number 17-M01-DTU.

Informed Consent Statement: Not applicable.

Acknowledgments: This work was partly funded by the Ministry of Foreign Affairs of Denmark and administered by the Danida Fellowship Centre through the 'Multi-scale and model-chain Evaluation of Wind Atlases' (MEWA) project. P.Q. acknowledges a tuition waver for M.Sc. studies from Tecnológico de Monterrey, as well as a stipend from CONACYT (Mexico). O.P. and G.C. appreciate support from the institutional research group on Energy and Climate Change at Tecnológico de Monterrey. J.L.P. is grateful to the institutional research group on Data Science and Optimization at Tecnológico de Monterrey. All authors acknowledge the wind resource data from an undisclosed wind project developer, without which, this study would not have been possible.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

$\#\{x\}$	The cardinality of the set x
$\hat{c}$	The estimated type of regression
$\hat{k}$	The estimated number of neighbors
$\hat{y}_i$	The estimated variable (uniform weighted regression)
$\hat{y}_i^{(w)}$	The estimated variable (inverse distance weighted regression)
$\mathcal{F}$	The set of all possible combination of features
μ_x	The mean of the variable x
$\overline{\overline{X}}$	The matrix of mean predictor values
$\overline{x_0}$	The vector of mean predictor values at site x_0
σ_x	The standard deviation of the variable x
c	The type of regression either uniform weighting or inverse distance weighting
c^*	The optimal type of regression
f_α	The set of features
K	The maximum number of neighbors
k	The number of neighbors
k^*	The optimal number of neighbors
L_2	The L_2 norm or Euclidean distance
MPE	The mean percentage error
$MPE^{(w)}$	The MPE obtained from a inverse distance weighing regression
$MPE_{site,\alpha,k}$	The MPE obtained for a given site from a set of features α and k number of neighbors
N	The number of observations
N_f	The number of features
$N_k(x_0)$	The set of nearest neighbors to the site x_0
$Sites$	The set of sites to be predicted
w	The weight of the regression either one or calculated with the inverse distance
X	The matrix of predictors
x_0	The target site
y_i	The observed variable to be predicted at a time step i

References

1. Lazard's Levelized Cost of Energy and Storage, v14, 2020. Available online: https://www.lazard.com/perspective/lcoe2020 (accessed on 14 March 2021).
2. Williams, J.H.; Jones, R.A.; Haley, B.; Kwok, G.; Hargreaves, J.; Farbes, J.; Torn, M.S. Carbon-Neutral Pathways for the United States. *AGU Adv.* **2021**, *2*, e2020AV000284. [CrossRef]
3. Buira, D.; Tovilla, J.; Farbes, J.; Jones, R.; Haley, B.; Gastelum, D. A whole-economy Deep Decarbonization Pathway for Mexico. *Energy Strategy Rev.* **2021**, *33*, 100578. doi:10.1016/j.esr.2020.100578. [CrossRef]
4. Bompard, E.; Grosso, D.; Huang, T.; Profumo, F.; Lei, X.; Li, D. World Decarbonization through Global Electricity Interconnections. *Energies* **2018**, *11*, 1746. [CrossRef]
5. Mortensen, N.G. *Wind Resource Assessment Using the WAsP Software Department of Wind Energy E Report 2019*; 2018; p. 45. https://backend.orbit.dtu.dk/ws/portalfiles/portal/164389714/Wind_resource_assessment_using_the_WAsP_software_DTU_Wind_Energy_E_0174_.pdf (accessed on 17 July 2021)
6. Beaucage, P.; Brower, M.C.; Tensen, J. Evaluation of four numerical wind flow models for wind resource mapping. *Wind Energy* **2014**, *17*, 197–208. [CrossRef]
7. Spyridonidou, S.; Vagiona, D.G. Systematic Review of Site-Selection Processes in Onshore and Offshore Wind Energy Research. *Energies* **2020**, *13*, 5906. [CrossRef]
8. Gomes Da Silva, A.F.; Peña, A.; Hahmann, A.N.; Zaparoli, E.L. Evaluation of two microscale flow models through two wind climate generalization procedures using observations from seven masts at a complex site in Brazil. *J. Renew. Sustain. Energy* **2018**, *10*. [CrossRef]
9. Palma, J.M.; Castro, F.A.; Ribeiro, L.F.; Rodrigues, A.H.; Pinto, A.P. Linear and nonlinear models in wind resource assessment and wind turbine micro-siting in complex terrain. *J. Wind Eng. Ind. Aerodyn.* **2008**, *96*, 2308–2326. [CrossRef]

10. Bechmann, A.; Sørensen, N.N.; Berg, J.; Mann, J.; Réthoré, P.E. The Bolund Experiment, Part II: Blind Comparison of Microscale Flow Models. *Bound. Layer Meteorol.* **2011**, *141*, 245–271. [CrossRef]
11. Hristov, Y.; Oxley, G.; Agar, M. Improvement of AEP predictions using diurnal CFD modelling with site-specific stability weightings provided from mesoscale simulation. *J. Phys. Conf. Ser.* **2014**, *524*. [CrossRef]
12. Veronesi, F.; Grassi, S.; Raubal, M.; Hurni, L. Statistical learning approach for wind speed distribution mapping: The uk as a case study. *Lect. Notes Geoinf. Cartogr.* **2015**, *217*, 165–180. [CrossRef]
13. Foresti, L.; Tuia, D.; Kanevski, M.; Pozdnoukhov, A. Learning wind fields with multiple kernels. *Stoch. Environ. Res. Risk Assess.* **2011**, *25*, 51–66. [CrossRef]
14. Lawan, S.M.; Abidin, W.A.; Masri, T. Implementation of a topographic artificial neural network wind speed prediction model for assessing onshore wind power potential in Sibu, Sarawak. *Egypt. J. Remote Sens. Space Sci.* **2020**, *23*, 21–34. [CrossRef]
15. Tang, X.Y.; Stoevesandt, B.; Fan, B.; Li, S.; Yang, Q.; Tayjasanant, T.; Sun, Y. An on-site measurement coupled CFD based approach for wind resource assessment over complex terrains. In Proceedings of the I2MTC 2018—2018 IEEE International Instrumentation and Measurement Technology Conference: Discovering New Horizons in Instrumentation and Measurement, Houston, TX, USA, 14–17 May 2018 pp. 1–6. [CrossRef]
16. Bechmann, A.; Leon, J.P.M.; Olsen, B.T.; Hristov, Y.V. The most similar predictor—On selecting measurement locations for wind resource assessment. *Wind Energy Sci.* **2020**, *5*, 1679–1688. [CrossRef]
17. Shabani, S.; Samadianfard, S.; Sattari, M.T.; Mosavi, A.; Shamshirband, S.; Kmet, T.; Várkonyi-Kóczy, A.R. Modeling Pan Evaporation Using Gaussian Process Regression K-Nearest Neighbors Random Forest and Support Vector Machines; Comparative Analysis. *Atmosphere* **2020**, *11*, 66. [CrossRef]
18. Mortensen, N.G.; Rathmann, O.; Tindal, A.; Landberg, L. Field validation of the ΔRIX performance indicator for flow in complex terrain. In Proceedings of the European Wind Energy Conference and Exhibition 2008, Brussels, Belgium, 31 March–3 April 2008; Volume 1, pp. 186–204.
19. Toledo, C.; Chávez-Arroyo, R.; Loera, L.; Probst, O. A surface wind speed map for Mexico based on NARR and observational data. *Meteorol. Appl.* **2015**, *22*, 666–678. [CrossRef]
20. Farr, T.; Rosen, P.; Caro, E.; Crippen, R.; Duren, R.; Hensley, S.; Kobrick, M.; Paller, M.; Rodriguez, E.; Roth, L.; et al. Shuttle Radar Topography Mission: Mission to map the world. *Rev. Geophys.* **2008**, *45*, 3–5. [CrossRef]
21. Earth Resources Observation Furthermore, Science (EROS) Center. Shuttle Radar Topography Mission (SRTM) 1 Arc-Second Global, 2017. Available online: https://doi.org/10.5066/F7PR7TFT (accessed on 17 July 2021).
22. Jun, C.; Ban, Y.; Li, S. Open access to Earth land-cover map. *Nature* **2014**, *514*, 434. [CrossRef] [PubMed]
23. Bontemps, S.; Defourny, P.; Bogaert, E.V.; Kalogirou, V.; Perez, J.R. GLOBCOVER 2009 Products Description and Validation Report. *ESA Bull.* **2011**, *136*, 53.
24. Floors, R.R.; Troen, I.; Kelly, M.C. Implementation of Large-Scale Average Geostrophic Wind Shear in WAsP12.1. 2018. p. 11. Available online: https://backend.orbit.dtu.dk/ws/portalfiles/portal/149165080/e_report_0169.pdf (accessed on 17 July 2021).
25. WindSim. Available online: http://windsim.com/ (accessed on 13 March 2021).
26. Rout, A.; Sahoo, S.S.; Singh, S.; Pattnaik, S.; Barik, A.K.; Awad, M.M. Chapter 7—Benefit-cost analysis and parametric optimization using Taguchi method for a solar water heater. In *Design and Performance Optimization of Renewable Energy Systems*; Assad, M.E.H., Rosen, M.A., Eds.; Academic Press: Cambridge, MA, USA, 2021; pp. 101–116. [CrossRef]
27. Chávez-Arroyo, R.; Gómez, H.; Herbert, F.J.; Romo Perea, A.; Probst, O. Mesoscale modeling and remote sensing for wind energy applications. *Rev. Mex. Física* **2013**, *59*, 114–129.
28. Basse, A.; Pauscher, L.; Callies, D. Improving Vertical Wind Speed Extrapolation Using Short-Term Lidar Measurements. *Remote Sens.* **2020**, *12*, 1091. [CrossRef]