Convolutional Encoding and Normalizing Flows: A Deep Learning Approach for Offshore Wind Speed Probabilistic Forecasting in the Mediterranean Sea

ROBIN MARCILLE,[a,b] PIERRE TANDEO,[b] MAXIME THIÉBAUT,[a] PIERRE PINSON,[c,d,e] AND RONAN FABLET[b]

[a] *France Énergies Marines, Technopôle Brest-Iroise, Plouzané, France*
[b] *IMT Atlantique, Lab-STICC, UMR CNRS 6285, Plouzané, France*
[c] *Imperial College London, Dyson School of Design Engineering, London, United Kingdom*
[d] *Department of Technology, Management and Economics, Technical University of Denmark, Lyngby, Denmark*
[e] *Halfspace, Copenhagen, Denmark*

(Manuscript received 22 December 2023, in final form 6 May 2024, accepted 28 June 2024)

ABSTRACT: The safe and efficient execution of offshore operations requires short-term (1–6 h ahead) high-quality probabilistic forecasts of metocean variables. The development areas for offshore wind projects, potentially in high depths, make it difficult to gather measurement data. This paper explores the use of deep learning for wind speed forecasting at an unobserved offshore location. The proposed convolutional architecture jointly exploits coastal measurements and numerical weather predictions to emulate multivariate probabilistic short-term forecasts. We explore both Gaussian and non-Gaussian neural representations using normalizing flows. We benchmark these approaches with respect to state-of-the-art data-driven schemes, including analog methods and quantile forecasting. The performance of the models and resulting forecast quality are analyzed in terms of probabilistic calibration, probabilistic and deterministic metrics, and as a function of weather situations. We report numerical experiments for a real case study off the French Mediterranean coast. Our results highlight the role of regional numerical weather prediction and coastal in situ measurement in the performance of postprocessing. For single-valued forecasts, a 40% decrease in RMSE is observed compared to the direct use of numerical weather predictions. Significant skill improvements are also obtained for the probabilistic forecasts, in terms of various scores, as well as an acceptable probabilistic calibration. The proposed architecture can process a large amount of heterogeneous input data and offers a versatile probabilistic framework for multivariate forecasting.

KEYWORDS: Operational forecasting; Probability forecasts/models/distribution; Deep learning

1. Introduction

Weather forecasting in offshore environments is challenging due to the scarcity and sparsity of offshore observations, in both space and time (Archer et al. 2014; James et al. 2018). These limitations affect data assimilation systems, especially initial state estimation, and validation processes. Moreover, wind profiles are challenging (Tambke et al. 2005) and influenced by various factors, such as air/sea exchanges (Optis et al. 2021), diurnal variations (Pichugina et al. 2017), and site-dependent effects (Sward et al. 2023), which are difficult to model accurately. Offshore weather forecasts are essential for marine operations, especially at locations where in situ data are scarce. These forecasts inform decision-making at sea for weather-limited operations. Weather operability limits are computed from simulation to avoid operation failure, and weather windows with critical parameters under the operability limits have to be forecast. Forecast errors imply risks of operation failure, and forecast uncertainty ought to be considered for operation planning and execution. To deal with uncertainty in the offshore wind energy industry, a factor ranging from 0 to 1 (the alpha factor) is assigned to each weather operability limit (Det Norske Veritas 2011). According to Gilbert et al. (2021), most existing methods rely on deterministic forecasts and the use of the alpha factor to account for weather forecast uncertainty. This may result in conservative decision-making and suboptimal planning. As illustrated in Gintautas and Sørensen (2017), probabilistic forecasts can address these shortcomings. Under the assumption of reliable weather forecast of the limiting parameters, the uncertainty can directly be transferred to the probability of operation failure. When doing so, one can obtain a large improvement in operational hours compared to the alpha-factor methodology. This requires the reliable joint probabilistic forecasting of limiting wind and wave parameters that impact vessel motions [e.g., significant wave height, 10-m wind speed, and wave peak period (Leontaris et al. 2016)]. The decision-making using probabilistic forecasts is then cost optimal compared to deterministic forecasts (Taylor and Jeon 2018; Catterson et al. 2016), motivating the development of probabilistic postprocessing of deterministic forecasts.

State-of-the-art weather forecasting systems generally rely on ensemble methods to assess and describe forecast uncertainty (Slingo and Palmer 2011). They generate different scenarios by varying both the initial state of the system and model parameters to estimate the spread of the forecast state. The very high computational cost associated with this forecast process limits the number of members in the ensemble, typically up to a few tens of members. Such ensembles cannot fully inform the forecasting uncertainties, especially for local processes such as strong convective events in the southeastern

Corresponding author: Robin Marcille, robin.marcille@france-energies-marines.org

DOI: 10.1175/AIES-D-23-0112.1 e230112

French maritime facade (Gulf of Lion) which is the main study area. The postprocessing of numerical weather prediction (NWP) using statistical and machine learning methods then appears appealing to better emulate these forecast uncertainties (Vannitsem et al. 2021).

A large variety of models can be used for the probabilistic postprocessing of deterministic forecasts (Bazionis and Georgilakis 2021). We can distinguish models based on the description of the probabilistic output. Nonparametric methods such as interval or quantile forecasting (Zou et al. 2022), kernel density, and ensemble methods make fewer assumptions about the shape of the target distribution. For instance, generalized additive model boosting for location, scale, and shape (gamboostLSS) and gradient boosting machine (GBM) can be used for the quantile forecasting of wave parameters (Gilbert et al. 2021). Parametric approaches assume a certain parametric distribution for the output (e.g., Gaussian, beta, and lognormal) (Afrasiabi et al. 2021) which allows for analytical computations. Within parametric descriptions, the Gaussian assumption might be simple but can characterize satisfyingly the uncertainty of two-dimensional wind prediction (Pinson 2012). One can estimate the parameters of Gaussian distribution using analogs of the observed weather situation (Lguensat et al. 2017; Platzer et al. 2021). Alternatively, regression and deep learning models can emulate a Gaussian covariance matrix from a deterministic forecast as developed in Sacco et al. (2022) considering a diagonal covariance matrix.

Novel generative deep learning techniques offer innovative methods for the approximation of complex posterior distributions. Variational recurrent autoencoders (VRAEs) can be used to generate scenarios at a relatively low computational cost (Zheng et al. 2022), but the output distribution can only be accessed via sampling. VRAEs are compared in Dumas et al. (2022) to generative adversarial networks (GANs) and normalizing flows for wind power forecasting. Normalizing flows are deep learning models based on the composition of parameterized bijective functions that transform a simple parametric distribution into an arbitrarily shaped distribution. It was proposed for variational inference in Rezende and Mohamed (2015) and generalized to density estimation in Dinh et al. (2017). Compared to analog methods, it needs no parametric assumption for the posterior distribution. In addition to sampling capabilities, they allow for exact likelihood computation. These two features are advantages compared to quantile forecasting. In contrast with VRAE and GAN, they are relatively easy to implement and train. In Rasul et al. (2021), conditional normalizing flows are shown to be well suited for multivariate time-series forecasting. A fair assessment of their advantages and disadvantages for a real application in probabilistic forecasting is lacking from the literature.

In light of the work cited above, this paper addresses the postprocessing of numerical weather prediction and in situ measurements using deep learning schemes to improve the probabilistic forecasting of wind speed at sea. Numerical weather prediction acts as a physical prior of the future state of the weather system at the considered offshore location, while recent neighboring measurements may better inform the actual state of the system. In this study, a parametric Gaussian model and a generative model using normalizing flows are compared with baseline models (analogs, gradient boosting machines, and numerical weather prediction) to analyze their performances in terms of probabilistic and deterministic metrics. Models are also compared as a function of the weather situation, to highlight the advantages and disadvantages of the method for marine operations. Eventually, the importance of various input data is discussed, to give indications on the required input data for offshore wind speed probabilistic forecasting.

The dataset used for the experiment is described in section 2. The proposed approach and its mathematical formalism are thoroughly presented in section 3, before the baseline methods and metrics used for comparison are detailed in section 4. The obtained results are shared and analyzed with deterministic and probabilistic metrics and as a function of weather situations in section 5. A discussion on the limitations of the experiment is done in section 6 to provide recommendations and perspectives for future work.

2. Dataset

a. *Case-study area*

To develop the methodology, we consider the MeteoNet dataset (Larvor et al. 2020). It is an open-source dataset developed and shared by Météo France, the French national weather service. It contains time series of weather ground station (GS) data and numerical weather prediction model over a 550 km × 550 km region in southeast France. It spans between 2016 and 2018 with 65 days of missing data. Hourly forecasts of weather variables (10-m wind speed, 2-m relative humidity, 2-m air temperature, and pressure at sea level) from the high-resolution model Applications de la Recherche à l'Opérationnel à Mesoéchelle (AROME) are available. AROME is the operational high-resolution model on France operated by Météo France. It has a grid size of 1.3 km and outputs hourly predictions. The ground station network covers 484 stations scattered over the southeast of France, as shown in Fig. 1a, with 6-min measured time series of 10-m wind speed, 2-m air temperature, station pressure, 2-m dewpoint temperature, 2-m relative humidity, and precipitation.

The study focuses on the Gulf of Lion, which is situated in the northeast Mediterranean Sea, between the cities of Toulon and Perpignan in southeast France. It is considered one of the main floating offshore wind development areas in France (Marcille et al. 2023). The study area is characterized by a strong dominance of offshore blowing winds in the northern (Mistral) and western (Tramontane) Gulf of Lion. Those phenomena are due to an orographic channeling in the Rhone and Garona valleys with the pressure difference between the northeast Atlantic (high pressures) and the northwest Mediterranean Sea (Gulf of Genoa, low pressure). When the high pressures are rather localized over central Europe, the region experiences strong southeast winds charged with humidity that can cause heavy precipitation on the coastal areas. These two phenomena are largely driving the wind patterns in the

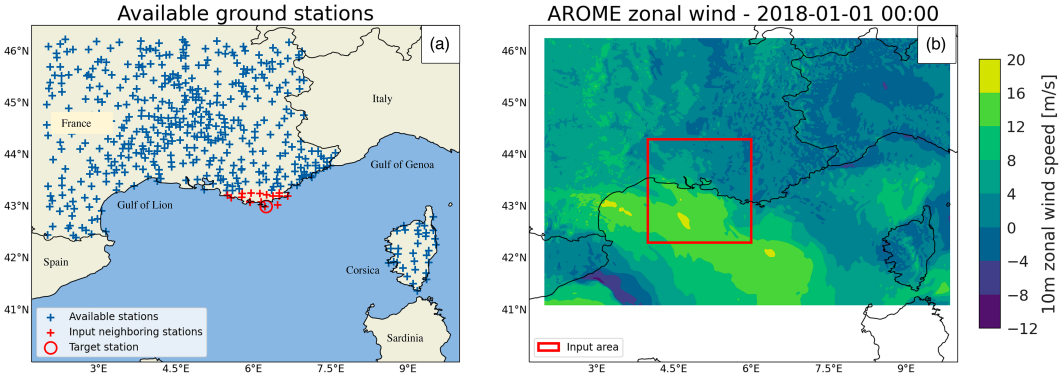

FIG. 1. Subset of the MeteoNet dataset selected for methodology development. (a) Coastal stations around Porquerolles target station are selected. (b) A geographical subset containing local information from NWP is selected to reduce the dimensionality of the input.

area and are sensitive to continental forcing and local orography. They also apply a strong forcing on the hydrodynamics of the region, with large upwelling and downwelling phenomena (Schaeffer et al. 2011).

The target station is the Porquerolles Island weather station encircled in Fig. 1a. It is the only offshore station available in the dataset. It is located on Porquerolles Island's semaphore, at 135 m of elevation on the top of the island. The 14 closest coastal weather stations in Fig. 1a are selected to serve as input. The numerical weather prediction input is reduced to a subset of 2° of latitude and longitude around the target station to reduce its dimensionality (see Fig. 1b). The correlation between the measured parameters at the input ground stations and the wind speed measured at the target station is shown in Fig. 2.

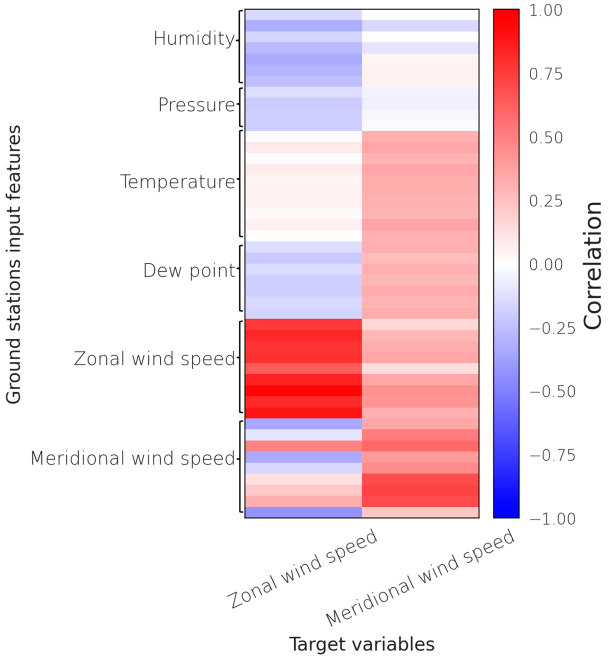

FIG. 2. Correlation between measured variables at ground stations and wind speed at the target station.

Wind speed at coastal ground stations is highly correlated with the target station. Zonal wind speed at the target station is negatively correlated with humidity at coastal stations, showing the predominance of eastern wind during rain events. Temperature is correlated with meridional wind, in link with thermal breezes.

b. *Numerical weather prediction data*

The numerical weather prediction input tensor at forecast issue time t, $\mathbf{X}_t^{\mathrm{NWP}}$ is a four-dimensional tensor in latitude (80 points), longitude (80 points), weather variables (five variables), and lead times (six time steps). The input variables available in the MeteoNet dataset are the two-dimensional 10-m wind speed (u, v), percentage of humidity, mean pressure at sea level, and 2-m temperature. The time step of the model data is 1 h, and the last forecast time step is $\tau_{\mathrm{NWP}} = 5$ h ahead. For each forecast issue time t, the AROME input has then $K_{\mathrm{NWP}} = 6$ lead times between t and $t + \tau_{\mathrm{NWP}}$. The variable and lead time dimensions are merged into a 30-dimensional axis, so the final tensor has dimensions (80, 80, 30). These data correspond to the deterministic forecast of AROME, with no information on the forecast uncertainty.

Correlation between AROME forecasts and wind speed at the target station is shown in Fig. 3. Lower pressures on the eastern part of the study area (Gulf of Genoa) are negatively correlated with zonal wind speed at the target station, showing the weather systems that channel Mistral northwestern winds. Higher correlations are observed for the zonal wind speed which is more representative of dominant wind systems. Meridional wind speed is more uncertain and is correlated with humidity and temperature.

c. *In situ data*

The input data from ground stations contain recent observations from the neighboring coastal stations. The ground station input tensor for the forecast issue time t, $\mathbf{X}_t^{\mathrm{GS}}$ is a three-dimensional tensor in stations (14 stations), weather variables (maximum six variables, depending on the station), and time steps (60 time steps). The input variables available at each station are the two-dimensional 10-m wind speed (u, v), humidity rate, temperature, pressure at sea level, and dewpoint.

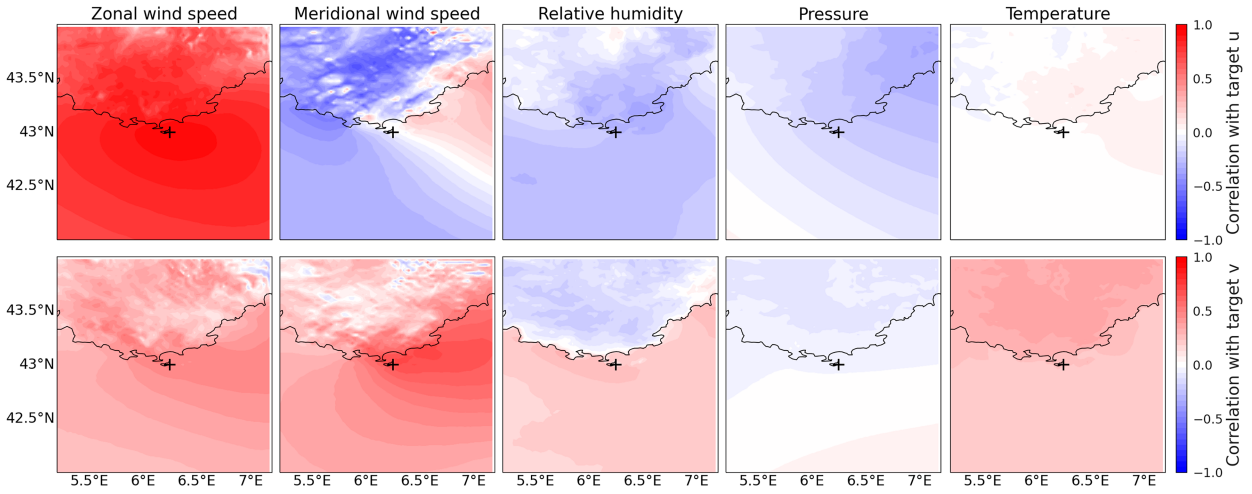

FIG. 3. Correlation between AROME forecasts used as input and wind speed at the target station. The correlation is computed for each grid point. (top) The correlation with the target zonal wind speed (target u), and (bottom) the target meridional wind speed (target v).

It has a time step of 6 min, and the last $\tau_{GS} = 6$ h of observations is used as input. The ground station input is then a concatenation of time series of $K_{GS} = 60$ time steps. The station and weather variable dimensions are merged so the final tensor has dimensions (80, 60).

The output of the dataset is the measured wind speed at the target station. At forecast issue time t, the target vector $\mathbf{y}_t$ is a tensor of zonal and meridional wind speeds at 10 m for different lead times. It has a time step of 6 min and is to be predicted for the next $\tau_{pred} = 6$ h. The target tensor consists in $N = 2$ time series of $K = 60$ lead times and has dimensions (2, 60).

To deal with missing data, measured variables from the ground stations exceeding 4% of missing data are removed. It corresponds to 3 weather stations and 34 measured weather variables in total. The resulting entries exceeding 4% of missing data are also removed (395 entries). Eventually, the remaining gaps in the data are forward filled.

d. *Training, testing, and validation datasets*

The dataset is split into three parts of training, validation, and testing phases. These three datasets need to be independent but representative of the same statistical distribution (Goodfellow et al. 2016). For weather data, autocorrelation at different time scales requires special care (Schultz et al. 2021). To limit seasonal effects, 2 years of data (two-thirds of the dataset) are used for training. The remaining third is split for validating and testing (half a year). Five days are removed in between the splits to avoid short-term temporal correlation between the datasets. To mitigate data representativity issues, cross validation on the train–validation–test split is performed. The train, validation, and test sets are shuffled into six different splits as shown in Fig. 4. Results are then computed across those six splits. After cleaning and splitting, the final dataset contains 2372 entries in the training split, 779 in the validation split, and 798 in the test split. All data sources are standardized with regard to the training dataset to ensure that all features have similar scales.

e. *Baseline reduced dataset*

The full dataset has a very high number of dimensions. To implement statistical baselines that can only accommodate a limited number of features, a baseline reduced dataset is constructed.

The reduced dataset contains the following:

- The three first principal components obtained through principal component analysis of both the zonal and meridional wind speeds of AROME inputs on the training dataset.
- The seven first principal components obtained through principal component analysis of the measured wind speed at the three closest ground stations for the last 6 h.
- The last wind measurements at the three closest ground stations.
- The wind speed forecast from the AROME closest grid point.

A sample from the reduced dataset $\mathbf{X}_t^r$ is then a tensor of 15 features and 60 lead times. The main dataset has dimensions (80, 80, 30) for AROME input and (46, 60) for ground station input. The reduced dataset corresponds to 0.5% of the total input data. Principal component analysis is used to extract the most relevant features.

The reduced dataset serves as input for the baseline methods presented below. It then allows for fair comparison between different approaches. The selected features of this reduced dataset were optimized to optimize the validation loss of the gradient boosting machine model in section 4c.

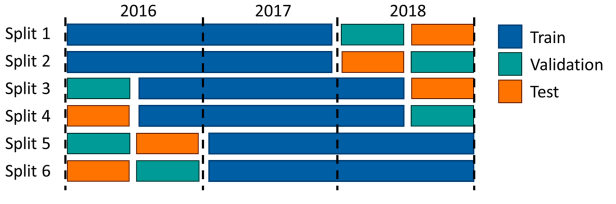

FIG. 4. Train validation test splits used for cross validation.

3. Proposed architecture

This section presents the proposed convolutional architecture to emulate a probabilistic multivariate forecast from the input data described in section 2. We first introduce the problem formulation in section 3a. The convolutional encoding of numerical weather prediction and in situ data is described in section 3b. We then detail the Gaussian (section 3c) and normalizing flow (section 3d) output probabilistic descriptions. Eventually, section 3e gives an overview of the final architecture.

a. Short-term wind forecasting at an unobserved location

The goal of the forecast model is to make a wind speed prediction at a target location using numerical weather prediction and ground station measurements. For a forecast issue time $t \in [1, T]$ and a forecast lead time $k \in [1, K]$, the model Ψ parameterized by Ξ outputs a vector $\hat{\boldsymbol{\theta}}_{t+k|t}$ from the input vector $\mathbf{X}_t$ such that

$$\hat{\boldsymbol{\theta}}_{t+k|t}(\Xi) = \Psi_{\Xi}(\mathbf{X}_t). \tag{1}$$

The output vector $\hat{\boldsymbol{\theta}}_{t+k|t}$ is a parameterization of a probability density function $\hat{f}_{t+k|t}$ of a random variable $S_{t+k|t}$ from which we can draw samples $\mathbf{s}_{t+k|t}$. The distribution $\hat{f}_{t+k|t}$ is transformed into a target distribution $\hat{g}_{t+k|t}$ through a transformation $\mathcal{T}$. Therefore, we map a sample $\mathbf{s}_{t+k|t}$ from the initial distribution into a sample $\mathbf{z}_{t+k|t}$ of the target distribution:

$$\mathbf{z}_{t+k|t} = \mathcal{T}(\mathbf{s}_{t+k|t}), \tag{2}$$

with $\mathbf{z}_{t+k|t}$ as a sample from the random variable $Z_{t+k|t}$ with probability density function $\hat{g}_{t+k}$. We explore an identity parameterization for transformation $\mathcal{T}$ as well as normalizing flows to account for more complex target distributions. In all that follows, the subscript k refers to $t + k|t$.

b. Convolutional encoding of AROME and ground station data

The proposed method uses a deep learning architecture to accommodate the large amount of heterogeneous input data. A convolutional neural network (CNN) is a type of deep neural network that uses convolutional layers and pooling layers to efficiently reduce the dimension of input data. Convolutional layers apply convolution filters to the input data, capturing multiscale features. The convolution filter applies the same weights to the whole input, so the number of model coefficients is reduced. Pooling layers reduce the dimension of the data by applying subsampling functions to groups of neighboring points (Goodfellow et al. 2016). CNNs are extensively used in the forecasting literature when dealing with large numerical model data in two dimensions (Obakrim et al. 2023) or three dimensions (Higashiyama et al. 2018). One-dimensional CNN can also be used to deal with time-series data (Zou et al. 2022).

For the offshore wind forecasting problem presented in this work, a large amount of data are used as input. Numerical weather prediction data are 80×80 images for each time step and each variable. Meteorological variables exhibit features at various scales that need to be extracted. A two-dimensional CNN is used to encode the numerical weather prediction input into an ensemble of latent time series containing useful information for forecasting. The convolutions are made through space to capture the spatial features, while the weather variables and lead times are taken as channels.

Seemingly, a one-dimensional CNN is used to encode the ground station time series onto a latent space. The convolution is performed on the time component, so that the temporal correlations of the time series can be captured. The 1D convolutional layers are used, and the different weather variables and stations are taken as channels.

We apply the CNN to numerical weather prediction and ground station time series to obtain 9 latent time series of 60 time steps. Two additional latent time series are added containing the predicted wind speed at the closest AROME grid point. The final dimension of the latent space is (11, 60).

c. Gaussian posterior assumption

The basic assumption for the proposed architecture describes the target as a two-dimensional Gaussian distribution. For a Gaussian posterior assumption, the output vector $\hat{\boldsymbol{\theta}}_k$ contains the parameters:

$$\hat{\boldsymbol{\theta}}_k = [\hat{\mu}_u(k), \hat{\mu}_v(k), \hat{\sigma}_u^2(k), \hat{\sigma}_v^2(k), \hat{\rho}_{u,v}(k)], \tag{3}$$

such that

$$Z_k \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k), \tag{4}$$

with $\hat{\boldsymbol{\mu}}_k$ as the mean matrix and $\hat{\boldsymbol{\Sigma}}_k$ as the covariance matrix, constructed from the two predicted variances $\hat{\sigma}_u^2(k)$, $\hat{\sigma}_v^2(k)$ and the Pearson coefficient $\hat{\rho}_{u,v}(k)$:

$$\hat{\boldsymbol{\Sigma}}_k = \begin{bmatrix} \hat{\sigma}_u^2(k) & \hat{\rho}_{u,v}(k)\hat{\sigma}_u(k)\hat{\sigma}_v(k) \\ \hat{\rho}_{u,v}(k)\hat{\sigma}_u(k)\hat{\sigma}_v(k) & \hat{\sigma}_v^2(k) \end{bmatrix}, \tag{5}$$

$$\hat{\boldsymbol{\mu}}_k = \begin{bmatrix} \hat{\boldsymbol{\mu}}_u(k) \\ \hat{\boldsymbol{\mu}}_v(k) \end{bmatrix}. \tag{6}$$

A 2-layer multilayer perceptron (MLP) is used to output Gaussian parameterization from the latent space. To ensure the positive semidefiniteness of the predicted covariance matrix, the variances should be positive $\sigma_u(k), \sigma_v(k) > 0$, and the Pearson coefficient should satisfy $-1 \leq \rho_{u,v}(k) \leq 1$. A final activation function is applied to the output of the MLP to satisfy these inequalities. The variances are obtained with the use of an exponential activation function, and the Pearson coefficient is obtained through a hyperbolic tangent activation function. The mean values $\mu_u(k), \mu_v(k) \in \mathbb{R}$ need no final activation function.

The loss function $\mathcal{L}_t(\Xi)$ used for the optimization is the negative log likelihood (Goodfellow et al. 2016):

$$\mathcal{L}_t(\Xi) = \frac{1}{K}\sum_{k=1}^{K} -\log[\hat{g}_{t+k|t}(\mathbf{y}_{t+k}|\Xi)], \tag{7}$$

with $\hat{g}_{t+k|t}$ as the predicted probability density function of the posterior assumption at lead time k. The negative log likelihood is a proper scoring rule that has two main advantages. It accounts for the reliability of the prediction defined through the covariance matrix, and it strongly penalizes outliers due to the log function.

Using a Gaussian distribution for the posterior assumption provides an analytical expression for the likelihood which can then be directly computed. For an observation $\mathbf{y}_k$ and a predicted two-dimensional Gaussian distribution with parameters $\hat{\boldsymbol{\Sigma}}_k$ and $\hat{\boldsymbol{\mu}}_k$, the likelihood is equal to (Goodfellow et al. 2016) the following equation:

$$\hat{g}_k(\mathbf{y}_k|\hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k) = \frac{1}{(2\pi)|\hat{\boldsymbol{\Sigma}}_k|^{1/2}} \times \exp\left[-\frac{1}{2}(\mathbf{y}_k - \hat{\boldsymbol{\mu}}_k)^T \hat{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{y}_k - \hat{\boldsymbol{\mu}}_k)\right]. \quad (8)$$

It is widely used for scoring forecasts versus observations under uncertainty for data assimilation schemes (Ruiz et al. 2022) and as a parametric method for multivariate regression (Muschinski et al. 2022).

d. Normalizing flows

A generative approach is proposed to account for non-Gaussian distributions while keeping the computation of the likelihood tractable and the sampling capabilities. Normalizing flows are generative deep learning models that use a composition of invertible functions to learn a "flow" from a simple base distribution (here a multivariate Gaussian) to an arbitrarily shaped distribution.

Given a base distribution $h^{(0)}$, and a series of invertible functions $\mathcal{T}_0$, ..., $\mathcal{T}_M$, the posterior likelihood can be computed using a change in variables from the base to the target distribution. The likelihood of the obtained distribution $h^{(M)}$ can then be obtained through a change in variable (Dinh et al. 2017):

$$\log[h^{(M)}(\mathbf{z}_M)] = \log[h^{(0)}(\mathbf{z}_0)] - \sum_{m=0}^{M} \log\left[\det\left|\frac{\partial h^{(m)}}{\partial \mathbf{z}_m}\right|\right]. \quad (9)$$

A sample from the base distribution is transformed into a sample from the target distribution using the following composition of transforms:

$$\mathbf{z}_M = \mathcal{T}_0 \circ \ldots \circ \mathcal{T}_M(\mathbf{z}_0). \quad (10)$$

A bijective function needs to be selected to compose the layers of the flow. In this work, a rational quadratic spline function is used. As described in Durkan et al. (2019), it has the advantage of being highly flexible while staying analytically invertible. Compared to more classical affine transformations, it can approximate complicated distributions with fewer transforms. The parameters of the transforms are the knot positions and the derivatives at each knot. These parameters are obtained through a 2-layer multilayer perceptron from the vector $\hat{\boldsymbol{\theta}}_k$.

Normalizing flows are implemented as an add-on block to the previously described architecture, so it transforms the predicted Gaussian distribution $\hat{f}_k = h^{(0)}$ into an arbitrarily shaped distribution $\hat{g}_k = h^{(M)}$ using $M = 5$ transforms. The transform applied to the Gaussian distribution $\hat{f}_k$ described in section 3c is then $\mathcal{T} = \mathcal{T}_0 \circ \ldots \circ \mathcal{T}_M$, and the set of parameters Ξ used for optimization contains the parameters of both the encoder and the normalizing flow block.

e. Final architecture

The final proposed architecture is shown in Fig. 5. It uses two convolutional encoders for numerical weather prediction data (three layers) and ground station data (two layers) to project the large amount of input data onto a latent space of dimension (13, 60). A multilayer perceptron of two fully connected layers is added with ReLU activation to obtain a time series of multivariate Gaussian distribution. Final care is given to ensure positive semidefiniteness for the covariance matrix with exponential and hyperbolic tangent activation functions for the correlation matrix.

To avoid overfitting, dropout layers are added to each of the two encoded blocks. The final model with Gaussian outputs has 2.6 million coefficients. Note that under the Gaussian posterior assumption, the predicted distribution $\hat{g}_k$ is equal to the Gaussian distribution $\hat{f}_k$.

The normalizing flow add-on block is trained together with the main architecture, transforming the predicted Gaussian multivariate distribution into an arbitrarily shaped distribution. The transformation is made for each time step and is composed of 10 layers parameterized with 1 fully connected layer of 128 hidden features. It adds 0.8 million parameters to the initial model.

The proposed architecture is named thereafter ConvE-STF for convolutional encoder for short-term forecasting. When considering a normalizing flow transformation, it is named ConvE-STF-NF.

All hyperparameters of the ConvE-STF and ConvE-STF-NF models were obtained using Bayesian optimization presented in section 4e to minimize validation loss.

4. Baselines and metrics

We describe below the state-of-the-art methods used as baselines to benchmark the proposed schemes. Considered performance metrics are detailed in section 4f.

a. Closest AROME grid point

The most straightforward baseline consists in considering the output of the AROME numerical weather prediction model at the closest grid point (i_c, j_c) from the target station. A linear regression computed on the training split is applied to the prediction:

$$\Psi^{\text{AROME}}(\mathbf{X}_t) = \frac{\mathbf{X}^{\text{NWP}}_{t,(i_c,j_c)} - \hat{\boldsymbol{\beta}}_0}{\hat{\boldsymbol{\beta}}_1}, \quad (11)$$

with $\hat{\boldsymbol{\beta}}_0$ and $\hat{\boldsymbol{\beta}}_1$ computed using ordinary least squares. The term $\mathbf{X}^{\text{NWP}}_{t,(i_c,j_c)}$ is the numerical weather prediction wind speed at the closest grid point from the target station. It is a deterministic output and is noted AROME in all that follows.

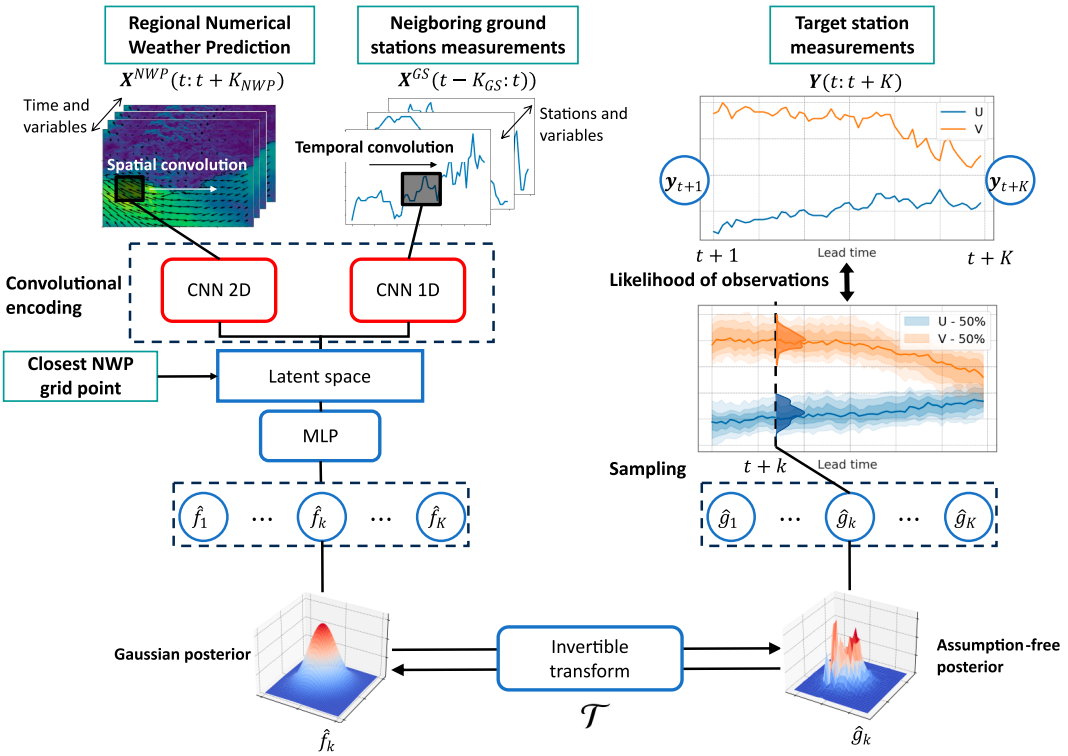

FIG. 5. Architecture of the ConvE-STF model illustrated for a forecast issue time t. Probabilistic forecast at the target station for lead times $t+1{:}\ t+k$ is obtained from NWP $\mathbf{X}_t^{\mathrm{NWP}} = \mathbf{X}^{\mathrm{NWP}}(t : t + K_{\mathrm{NWP}})$ and recent neighboring ground station measurements $\mathbf{X}_t^{\mathrm{GS}} = \mathbf{X}^{\mathrm{GS}}(t - K_{\mathrm{GS}} : t)$. A convolutional encoder outputs a time series of multivariate Gaussian distributions $\{\hat{f}_k\}_{k\in[1,K]}$ that are passed through an invertible transform $\mathcal{T}$ to output the predicted posterior distributions $\{\hat{g}_k\}_{k\in[1,K]}$.

b. *Analog forecasting*

Analog forecasting is a simple yet efficient statistical method for the forecasting of dynamical systems with unknown dynamics (Lguensat et al. 2017). From a catalog of past trajectories, analog situations are looked for according to a certain distance metric. The D nearest analogs of the current situation are selected, and their trajectories are considered as possible future scenarios. The analogs are weighted according to their distances to the target situation, and then, mean and covariance matrices are estimated from the ensemble of trajectories under the Gaussian assumption.

In this work, the distance metric in the catalog is the Minkowski norm on the variables of the reduced dataset. The weighting of the trajectories and the estimation of the Gaussian distribution are done under locally constant assumption using $D = 12$ analogs [see, e.g., Lguensat et al. (2017) and Platzer et al. (2021)]. Hyperparameters of the analog model were tuned with Bayesian optimization to minimize the validation loss.

c. *Gradient boosting machine*

Gradient boosting machines are tree-based regression methods that train an ensemble of weak-learner regression trees to perform a multiple nonlinear regression between output and input. Such methods are implemented in Gilbert et al. (2021) to create probabilistic significant wave height forecasts for offshore wind turbine access forecasting.

The gradient boosting algorithm uses the steepest descent algorithm to optimize the ensemble of regression trees according to a given loss function (Friedman 2001). Hyperparameters are the number of regression trees, the number of splits for each tree, and a shrinkage parameter that controls the weight of each tree in the ensemble. These parameters were tuned with Bayesian optimization to minimize the validation loss. In this work, a gradient boosting machine is trained with the quantile loss for each predicted quantile $\alpha \in Q = \{0.05, 0.15, \ldots, 0.45, 0.5, 0.55, \ldots, 0.85, 0.95\}$, variable $n \in [1, N]$, and lead time $k \in [1, K]$. For a two-dimensional output, the full model then consists in 1320 individual models. The predicted quantiles form a marginal quantile function for the two output parameters for each lead time. The obtained model is noted Ψ^{GBM} and referred to as GBM:

$$\Psi^{\mathrm{GBM}}(\mathbf{X}_t^r) = \{\Psi^{\mathrm{GBM}}_{\alpha,n,k}(\mathbf{X}_t^r)\}_{\alpha\in Q, n\in[1,N], k\in[1,K]}. \tag{12}$$

Overall, the output of each individual gradient boosting machine model contains the quantile prediction $\Psi^{\mathrm{GBM}}_{\alpha,n,k}(\mathbf{X}_t^r) = \hat{q}_{\alpha,n,k}$ for a specific quantile α, variable n, and lead time k.

For each time step and variable, we approximate the quantile function from the quantiles of the distribution. In addition

to second-order derivative continuity at the predicted knots, the monotony of the quantile function needs to be preserved. Cubic spline interpolation is then used (Fritsch and Carlson 1980; McKinley and Levine 1998) to obtain the quantile function from the predicted knots. It is a commonly used assumption for quantile function smoothing (Gilbert et al. 2021; He et al. 2021). Samples can then be drawn from this approximate quantile function to compute scores and generate scenarios. The quantile probabilistic description has the advantage of being assumption free on the shape of the posterior distribution. However, there is no explicit formulation for the likelihood of the distribution and quantile crossing can appear. It also has a substantial computational cost by requiring one model per quantile, variable, and lead time. It has no explicit control for overfitting, as it is only controlled by the hyperparameters of the fitting of regression trees.

d. *ConvE-STF-reduced*

To compare the statistical baselines with the proposed architecture, an additional baseline model is added. It consists of a similar convolutional architecture as the one of the proposed models in section 3, but runs with the reduced dataset described in section 2e as input. This reduced baseline is noted as ConvE-STF-reduced. Its hyperparameters are tuned using Bayesian optimization to minimize the validation loss.

e. *Hyperparameter tuning*

We tuned the hyperparameters of the different models presented in the following sections, and those of the reduced dataset in section 2e, using a Bayesian optimization framework (Akiba et al. 2019) with the loss metric on the validation dataset as optimization metric. Using the Python package Optuna (Akiba et al. 2019), it relies on tree-structured Parzen estimators (Bergstra et al. 2011) to retrieve optimal hyperparameters within a predefined search space. This Bayesian optimization applies to the following hyperparameters for the ConvE-STF model: kernel size, pool size, number of convolutional layers, dropout rates, latent space dimensions, number of fully connected layers, number of neurons in the fully connected layers, learning rate, weight decay, learning rate, decay rate, and batch size. For the gradient boosting machine, it is applied to reduced dataset features, learning rate, number of trees, maximum depth, minimum leaf samples, and minimum split samples. For the analogs, it is applied to the number of analogs, distance metric, and regression mode. Eventually, for the normalizing flows, the number of layers, number of hidden features, number of spline function bins, and dropout rate are optimized.

f. *Evaluation metrics*

Forecast quality is evaluated using an ensemble of deterministic and probabilistic metrics (Messner et al. 2020). Deterministic metrics compare the mean or median of the predicted distribution with observations. The mean value of the predicted distribution $\hat{f}_{t+k|t}$ is $\overline{\hat{\mathbf{y}}}_{t+k|t}$, and the median value is $\tilde{\hat{\mathbf{y}}}_{t+k|t}$. The root-mean-square error (RMSE) and the mean absolute error (MAE) are used in this work. Both metrics do not penalize outliers as strongly. The metrics are computed for each lead time k and noted with a subscript k when given as such. Global metrics across the dataset are averaged over all lead times and are noted without subscripts:

$$\mathrm{RMSE}_k = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\mathbf{y}_{t+k} - \overline{\hat{\mathbf{y}}}_{t+k|t})^2}, \tag{13}$$

$$\mathrm{MAE}_k = \frac{1}{T}\sum_{t=1}^{T}|\mathbf{y}_{t+k} - \tilde{\hat{\mathbf{y}}}_{t+k|t}|. \tag{14}$$

For probabilistic forecasts, the full predicted distribution should be scored against the observations. The continuous ranked probability score (CRPS) is a proper scoring rule for evaluating the performance of a distribution versus observations (Gneiting and Raftery 2007). It is a univariate score that is computed for each variable $n \in [1, N]$ and noted with a subscript n for the variables. The global score is averaged across variables and noted without subscript. The CRPS integrates the difference between the predicted cumulative density function and the indicator function at the observation value as follows:

$$\mathrm{CRPS}_n = \frac{1}{T}\frac{1}{K}\sum_{t=1}^{T}\sum_{k=1}^{K}\int_{-\infty}^{+\infty}[\hat{F}_{t+k|t}(y) - \mathbf{1}(y \le y_{t+k})]^2 dy. \tag{15}$$

When the cumulative density function is not tractable, the CRPS can be computed from samples drawn from the distribution. Gneiting and Raftery (2007) show that the CRPS can be computed from an ensemble of L samples as

$$\mathrm{CRPS}_n = \frac{1}{T}\frac{1}{K}\sum_{t=1}^{T}\sum_{k=1}^{K}\left\{\frac{1}{L}\sum_{l=1}^{L}\left[\left|y^n_{t+k} - \hat{y}^{n,(l)}_{t+k|t}\right|\right] - \frac{1}{2L^2}\sum_{l=1}^{L}\sum_{m=1}^{L}\left(\left|\hat{y}^{n,(l)}_{t+k|t} - \hat{y}^{n,(m)}_{t+k|t}\right|\right)\right\}, \tag{16}$$

with $\hat{y}^{n,(l)}_{t+k|t}$ as a sample $l \in [1, L]$ from the predicted distribution of variable n and y^n_{t+k} as the corresponding observation. The CRPS is equivalent to the MAE for deterministic forecasts (Messner et al. 2020).

The energy score (ES) is the multivariate generalization of the CRPS and can be computed from samples seemingly to Eq. (16) such that

$$\mathrm{ES} = \frac{1}{T}\frac{1}{K}\sum_{t=1}^{T}\sum_{k=1}^{K}\left\{\frac{1}{L}\sum_{l=1}^{L}\left[\left\|\mathbf{y}_{t+k} - \hat{\mathbf{y}}^{(l)}_{t+k|t}\right\|\right] - \frac{1}{2L^2}\sum_{l=1}^{L}\sum_{m=1}^{L}\left[\left\|\hat{\mathbf{y}}^{(l)}_{t+k|t} - \hat{\mathbf{y}}^{(m)}_{t+k|t}\right\|\right]\right\}, \tag{17}$$

with $\|\cdot\|$ as the Euclidian norm. CRPS and ES are mostly sensitive to the first moments of the distributions (Pinson and Girard 2012) so the variogram score (VS) is introduced. It only scores the correlation structure between the predicted variables and ignores the bias. It can be computed from samples as

TABLE 1. Probabilistic and deterministic metrics of implemented forecast models. The best obtained scores are shown in bold. The bracket scores show the MAE which is equivalent to the CRPS for deterministic forecasts. The scores are given as mean and standard deviation over the six splits.

Model	RMSE (m s^{-1})	CRPS (m s^{-1}) [MAE (m s^{-1})]	ES (m s^{-1})	VS
AROME	2.60 ± 0.04	[1.98 ± 0.02]	—	—
Analogs	2.23 ± 0.09	1.20 ± 0.05	1.89 ± 0.07	0.61 ± 0.01
GBM	1.93 ± 0.04	1.05 ± 0.02	1.65 ± 0.03	0.52 ± 0.01
ConvE-STF-reduced	1.93 ± 0.04	1.04 ± 0.02	1.65 ± 0.03	0.53 ± 0.02
ConvE-STF	1.57 ± 0.04	0.84 ± 0.02	1.31 ± 0.04	**0.39 ± 0.01**
ConvE-STF-NF	**1.56 ± 0.07**	**0.82 ± 0.04**	**1.29 ± 0.06**	**0.39 ± 0.02**

$$\mathrm{VS}_p = \frac{1}{T}\frac{1}{K}\sum_{t=1}^{T}\sum_{k=1}^{K}\left\{\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\left|y^i_{t+k} - y^j_{t+k}\right|^p - \frac{1}{L}\sum_{l=1}^{L}\left|\hat{y}^{i,(l)}_{t+k|t} - \hat{y}^{j,(l)}_{t+k|t}\right|^p\right]^2\right\}, \tag{18}$$

with p as the order of the variogram. It is set to 0.5 as recommended by Messner et al. (2020), and the score $\mathrm{VS}_{0.5}$ is noted as VS for simplicity.

Eventually, rank histograms are used to assess the reliability of the forecasts (Candille and Talagrand 2005). A probabilistic forecast is reliable if it predicts probabilities that fit with the observed relative frequencies. In the rank histogram, the quantiles in which the observations fall are counted. For an infinite number of observations, $1/Q$ of it should fall in the $\alpha \in Q$ quantile. The frequency of observed observations is displayed as bar plots, and a perfectly reliable forecast should display a flat rank histogram (i.e., uniform distribution). The multivariate generalization of the univariate rank histogram can be found in Gneiting et al. (2008).

The rank histogram is quantitatively evaluated thanks to the reliability index that measures the mean deviation of the bins to the perfect reliable model. With $\hat{b}_j$ as the frequency of observation falling below the jth predicted quantile $\hat{\alpha}_j$, the reliability index is defined as

$$\mathrm{REL} = \frac{1}{Q}\sum_{j=1}^{Q}\left|\hat{b}_j - \frac{1}{Q}\right|. \tag{19}$$

5. Results

a. Forecast evaluation

Table 1 shows the scores obtained by the different forecast models, with the best values shown in bold. All implemented methods improve the RMSE compared to the AROME forecast, showing the necessity to postprocess the output of numerical weather prediction models for a specific target station.

Baseline models using the reduced dataset as input are all skillful at postprocessing the numerical weather prediction with, for instance, a 26% decrease in RMSE for the gradient boosting machine forecast. The analog forecast also improves by 14% of the RMSE, with a higher variability. The proposed ConvE-STF architecture largely outperforms the gradient boosting machine by 0.36 m s^{-1} in RMSE and 0.21 m s^{-1} in CRPS, achieving a 40% reduction in RMSE compared to AROME. The ConvE-STF-reduced forecast is just as good as the gradient boosting machine model but is largely surpassed by the ConvE-STF model using the full input. It highlights the presence of explanatory variables in the input dataset and illustrates the capabilities of deep learning architecture to process a large amount of heterogeneous input. The ConvE-STF is 25% better than the gradient boosting machine at predicting the correlation structure between the outputs as shown by the VS, showing that the Gaussian description is competitive for the two-dimensional wind probabilistic forecast. Eventually, the ConvE-STF with normalizing flow block slightly improves the scores of the Gaussian output, with a higher variability between splits.

The evolution of the generalized RMSE as a function of lead time is shown in Fig. 6. Whereas the error clearly increases with the lead time for the AROME baseline, it is not exactly the case for the other models, for which the error stagnates or even decreases for the first 4 h of forecast. This is

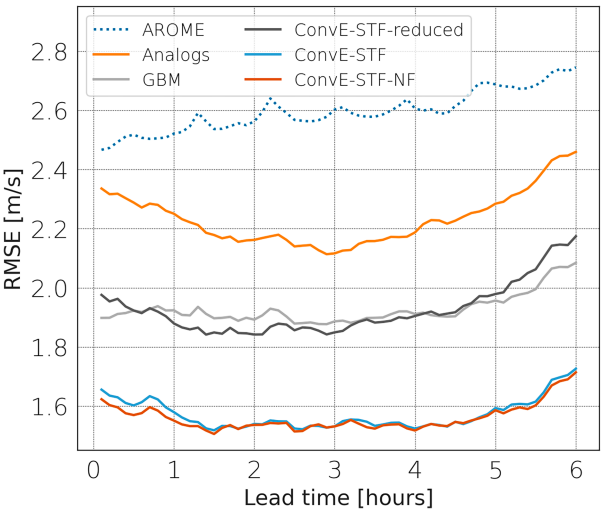

FIG. 6. Evolution of the RMSE for all models as a function of lead time. The top dashed line is the output of the NWP AROME corrected.

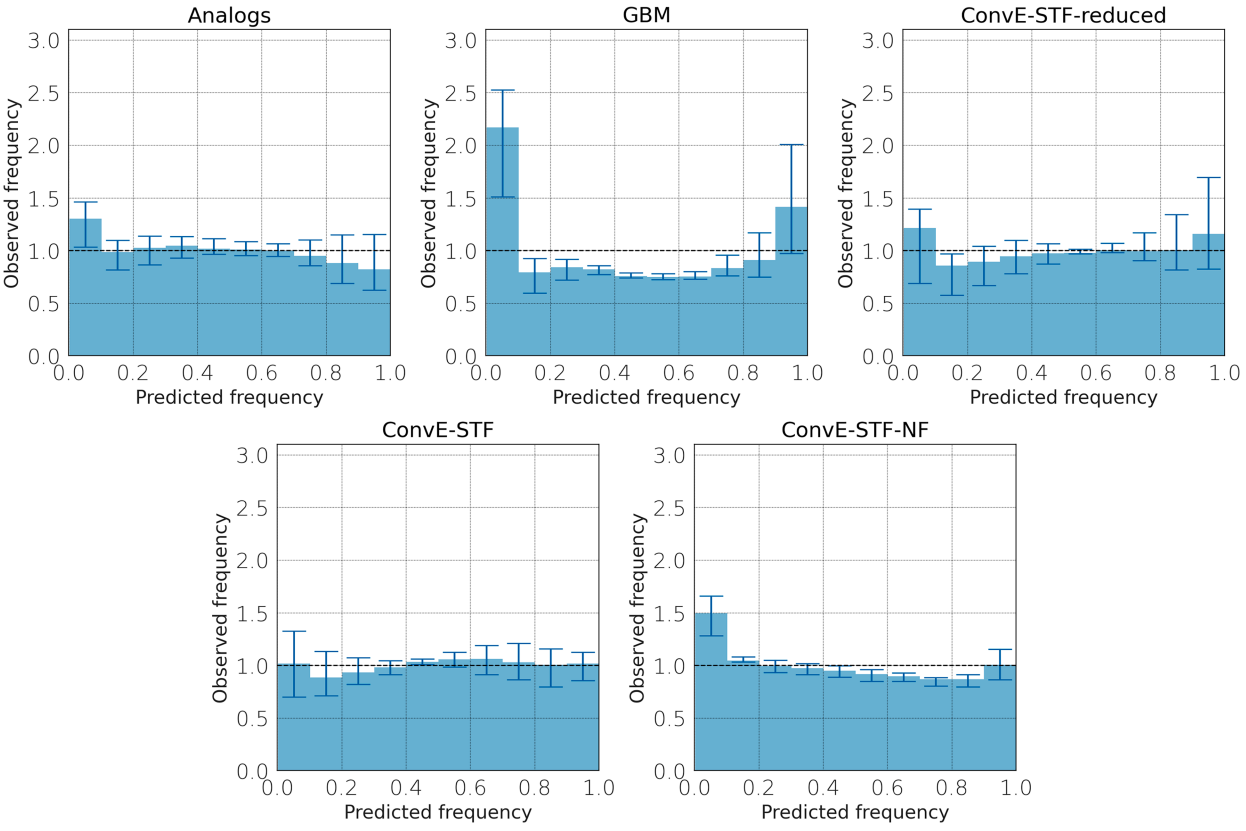

FIG. 7. Generalized 2D rank histograms obtained on the test set. Perfect model calibration is shown as a dashed black line.

likely to be due to diurnal effects coupled with fixed forecast issue times (0600, 1200, 1800, 2400 LT). The trend is not visible in the AROME baseline and is equally captured by analogs, gradient boosting machines, and ConvE-STF methods. It shows that it is a trend in the dataset, independent of input data. The proposed approach largely outperforms all baselines for all lead times.

The spatiotemporal correlation between the neighboring stations and the target station helps correcting the numerical weather prediction in the very short term. The ConvE-STF model, with its ability of ingesting a large amount of input data, shows a significant improvement throughout the forecast window.

b. *Reliability*

In Fig. 7, the observed quantiles are plotted versus the predicted quantiles as a rank histogram for all the forecast models. The dashed line represents a perfectly reliable forecast. The rank histogram is computed for each train–validation–test split, and the 50% interquantile range between splits is shown as error bars. The gradient boosting machine and ConvE-STF-reduced models show clear U-shaped rank histogram, which shows underdispersion (i.e., an underestimation of the uncertainty). The analog model, while showing poor deterministic and probabilistic quality metrics, is reliable though slightly overdispersive. Indeed, the analogs estimate a Gaussian distribution from existing trajectories, which guarantee a certain stability in the uncertainty estimation. However, the limited size of the catalog used can explain the overdispersion. The ConvE-STF and ConvE-STF-NF reliability is even more acceptable, with a slight difference for extreme quantiles. The difference in reliability between ConvE-STF and ConvE-STF-reduced shows that the choice of input data is of greater importance for forecast reliability than the choice of the posterior distribution. The ConvE-STF-NF and ConvE-STF achieve relatively similar reliability patterns with different posterior assumptions but the same input data and similar architectures.

The generalized reliability index is given in Table 2 to quantitatively assess the models' reliability. The very high variability with cross validation shows the sensitivity of models' reliability to the training dataset. It highlights the limitations of the obtained models due to dataset length. The ConvE-STF-NF

TABLE 2. Generalized reliability index for all models. Best obtained reliability index is shown in bold.

Model	Reliability index
Analogs	1.6 ± 0.5
GBM	3.2 ± 0.5
ConvE-STF-reduced	2.1 ± 1.2
ConvE-STF	1.5 ± 0.6
ConvE-STF-NF	**1.4 ± 0.4**

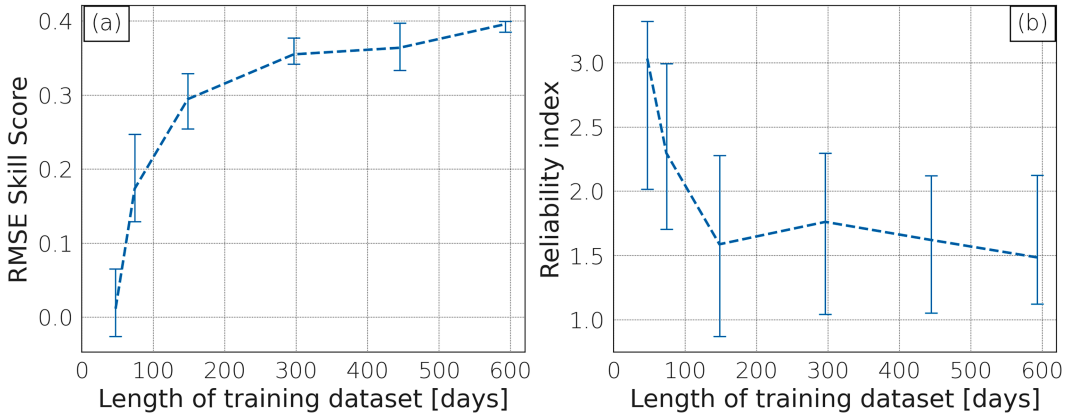

FIG. 8. Model improvement with training dataset length obtained with cross validation. (a) RMSE skill score vs AROME forecast and (b) reliability index.

model is the most reliable model with a reliability index of 1.4 and the lowest variability. The use of normalizing flows improves the model's reliability, showing the interest of relaxing the posterior parametric assumption for multivariate probabilistic forecast.

c. *Data representativity*

This study relies on a 33-month-long dataset to develop and benchmark deep learning–based postprocessing models. Following similar previous studies (Zheng et al. 2022; Gallego et al. 2011; Wang et al. 2017), we aim to assess the potential impact of the length of training dataset on the generalization performance of the trained models. We then train and assess the proposed ConvE-STF models using training datasets of different lengths from 2 months to 2 years as illustrated in Fig. 8. Overall, we observe the expected trend, the longer the training dataset, the better the model performance. In Fig. 8a, the RMSE skill score shows that from 60-day-long training datasets, we train ConvE-STF models which are more skillful than the AROME forecast. We also note a slower improvement of the forecasting skills from 1-yr-long datasets, as well as a lower variability between cross-validation splits. Similar results are observed for the model reliability in Fig. 8b. These results support the relevance of training datasets covering at least 1 or 2 years to retrieve a robust average improvement through the ConvE-STF models of the AROME forecasts.

d. *Computational cost*

The computational cost of the different models was evaluated for training, inference, and sampling. The deep learning models (ConvE-STF, ConvE-STF-NF, and ConvE-STF-reduced) are trained on a single 32Go NCIDA RTX A6000 graphics processing unit (GPU). The gradient boosting model is trained on multiple (60) AMD EPYC 7763 CPU. The obtained computational costs are given in Table 3.

The training of a gradient boosting machine for quantile forecasting requires the training of a single model for each variable, lead time, and quantile. In this study, this results in 1320 individual models. This results in a heavy model file (348 hPa) and implies multi-CPU training. The training time is then $O(NTQ)$, with N as the number of samples, T as the number of predicted lead times, and Q as the number of quantiles. Deep learning models are easily parallelized using GPU, resulting in a training time of ≈500 s for ConvE-STF on a single GPU. The addition of normalizing flows implies transformation inversion that adds computational cost for error gradient backpropagation, making it 3 times slower to train than ConvE-STF. Analog methods need no training time, making it very simple to implement probabilistic forecast framework. The sampling from the predicted distributions is more efficient under the Gaussian assumption. Normalizing flow transformation makes it 300 times slower than with a simple Gaussian posterior assumption, and the sampling using the empirical quantile function for the gradient boosting machine is 2000 times slower.

e. *Probabilistic wind speed forecasts*

The quantile description output by the gradient boosting machine is flexible as it makes no assumption on the underlying distribution. It can in theory capture heavy tail or multimodal

TABLE 3. Computational cost comparison.

Model	Machine	Model size (Mo)	Training CPU/GPU time (s)	Inference time (s)	Sampling time (s)
Analogs	CPU	0	0	0.076	0.076
GBM	60 CPU	348	3600	0.0005	1.05
ConvE-STF-reduced	GPU	1	45 (0.4 s epoch^{-1})	0.008	0.003
ConvE-STF	GPU	10	500 (1.4 s epoch^{-1})	0.015	0.006
ConvE-STF-NF	GPU	16	3500 (4.3 s epoch^{-1})	0.018	0.18

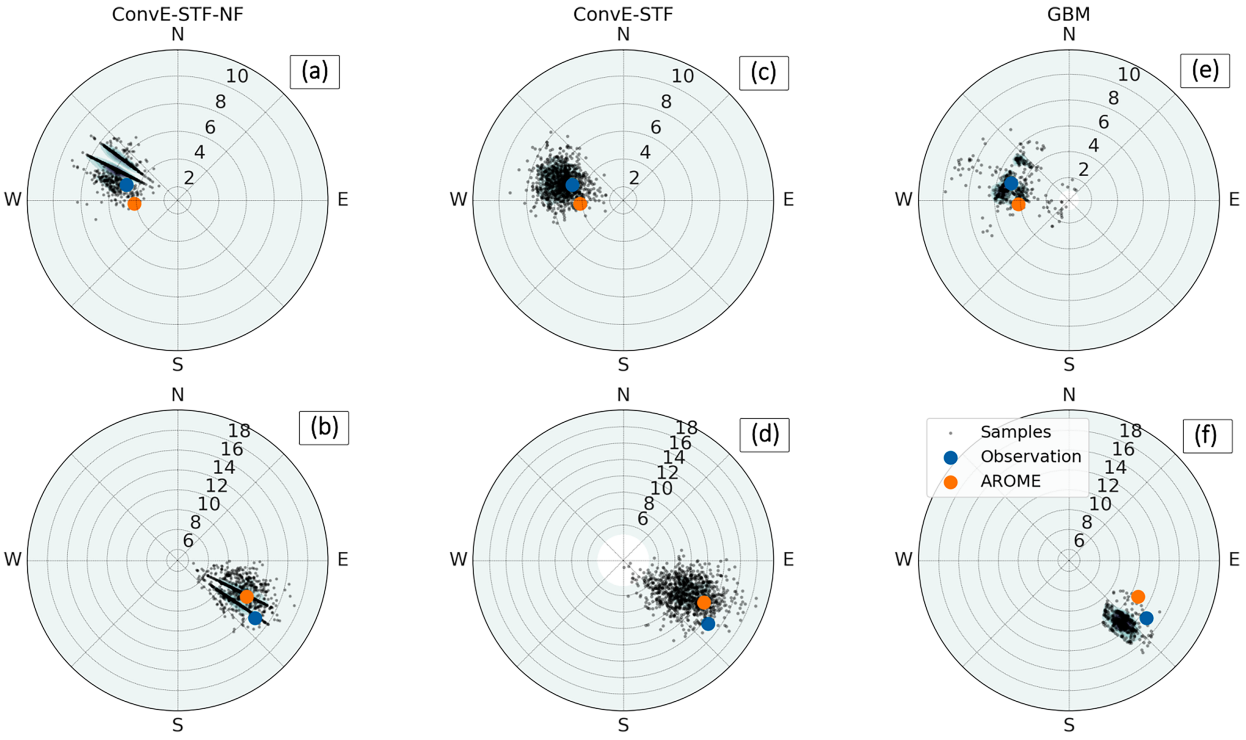

FIG. 9. The three different probabilistic approaches are illustrated on two entries of the dataset. Samples generated from the predicted distribution in the generative case (ConvE-STF-NF), Gaussian case (ConvE-STF), and quantile case (GBM) are scattered on polar plots of wind speed and wind direction. The observation is shown as a blue circle and the AROME prediction as an orange circle.

distribution. However, it is limited to the prediction of marginal distributions, and the correlation structure is not explicitly described. This is observed with the VS in section 5a. The lack of correlation structure in the gradient boosting machine output is a drawback for the joint probabilistic forecasting of correlated variables. For the two-dimensional wind speed, it can result in unrealistic sampled wind direction. Though it is hard to measure the impact of the correlation structure with standard statistical metrics, it is expected to strongly impact the generation of multivariate scenarios for offshore wind operation weather window forecasting.

In the ConvE-STF and analog methods, a multivariate Gaussian assumption is made for the output with $(2, 2)$ covariance matrices. The Gaussian assumption can be relaxed using normalizing flows in the ConvE-STF-NF model, but no clear quantitative effects are observed in terms of model performance. However, the normalizing flow approach adds little computational cost to the previous Gaussian assumption. By construction, the likelihood can be easily calculated, and samples can be directly generated. It can in theory adapt to complicated posterior distributions with a limited added model complexity. A sample from the latent Gaussian distribution is passed through several layers of neural splines (Durkan et al. 2019) to be transformed into a sample in real space. The nonlinearities within the neural spline flows can approximate very complex distributions and are conditioned by the input data. By doing so, we lift any assumption on the posterior data, compared to the quantile approach or the Gaussian assumption.

The shapes of the predicted distributions from the different methods are illustrated in Fig. 9 for two entries in the test dataset. For the first entry (Figs. 9a,c,e), the gradient boosting machine distribution has heavy tails, showing the flexibility of the quantiles. For the second entry (Figs. 9b,d,f), it has a very low spread, probably due to overfitting. Figures 9a and 9b show multimodal distributions obtained with normalizing flows. The obtained shapes are not very different from the Gaussian distributions in Figs. 9c and 9d, but show a discretization in wind direction. This is an artifact of the dataset, knowing that the wind direction at the target station is measured with a resolution of 5°. Normalizing flows can partially capture this complicated relationship between the predicted variables in a nonsupervised way. It shows the great flexibility of normalizing flows for probabilistic forecasting.

f. Input sensitivity

The ConvE-STF is trained with different input sets to compare the value of each data source. The size of the numerical weather prediction domain and the number of neighboring stations are the two main parameters considered for sensitivity. They are crucial parameters for the method generalization, and they can give indications on explanatory variable importance.

In Fig. 10a, the sensitivity of RMSE to the number of ground stations used as input is plotted. A clear trend is identified, with a decreasing RMSE for the 12 closest stations and a stabilization for an increased number of stations. This

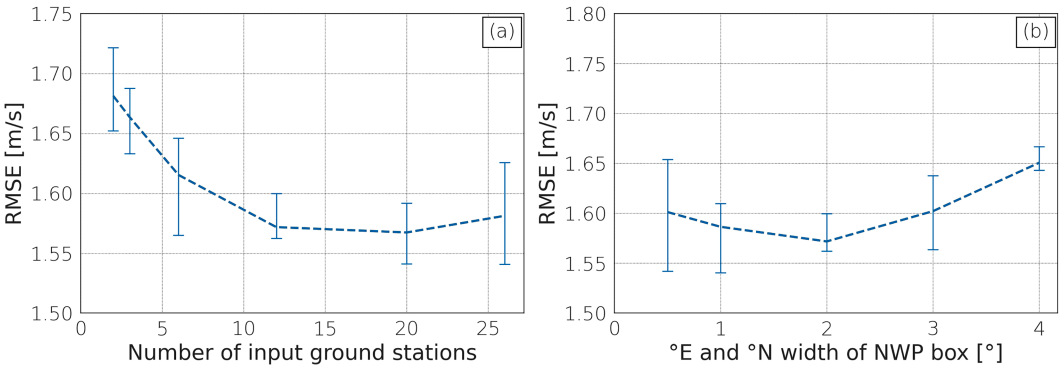

FIG. 10. Sensitivity of the RMSE (a) to the number of ground stations taken as input and (b) to the size of the NWP input. The dashed line is the generalized RMSE, and blue error bars show the 50% interquantile range over the six splits.

validates the choice of 12 closest stations as input for the main model. This optimal number of input stations, however, strongly depends on the experimental setup. First, it is site dependent and represents the limit of spatiotemporal correlation between measured parameters and target parameters. This is a function of the distance and position of the neighboring ground stations, which will be specific for every site. Second, it depends on the length of the time series considered as input. In this study, we limited the length of the neighboring measurement time series to 6 h. Longer time series might then exhibit larger areas of spatiotemporal correlation. Eventually, it depends on the length of the forecast window, which is for this experiment limited to 6 h.

In Fig. 10b, the sensitivity to the size of the input numerical weather prediction mask is shown. The change in input size (from 20 × 20 images to 120 × 159 images) implies a change in the convolutional architecture (from two to three layers). A hyperparameter tuning for the numerical weather prediction data encoder was made for each input size using Bayesian optimization as described in section 4e. The link between forecast error and numerical weather prediction input size is not as straightforward and can only be discussed for this specific site. The best performances are obtained with a mask of 2° in latitude and longitude. It is possible that the larger input area in this specific region does not carry more information than the smaller input mask, but there is no guarantee that even larger masks would not bring additional information. In particular, the atmospheric circulation in the eastern Gulf of Lion is notably influenced by the situation in the Gulf of Genoa and Ligurian Sea which would require a wider input mask.

In Fig. 11, we report the performance of ConvE-STF models using different combinations of wind data as inputs. We consider three wind data sources: namely, the wind measurements from ground station input (GS), the wind prediction from the operational NWP for the considered domain, and the wind prediction from the operational NWP for the grid point the closest to the targeted offshore location (Closest) (see Fig. 5). These results illustrate the relative importance of the different data sources in the prediction of the ConvE-STF model. The addition of ground station input greatly improves the RMSE compared to the two central bars. It highlights the importance of neighboring measurements as explanatory variables. From the GS-only case, it can be noted that both the addition of numerical weather prediction input and closest grid point input improve the forecast postprocessing. It shows that information can be extracted from regional forecasts to improve the forecast at a target station, but that it is hard to capture the forecast at the closest grid point using convolution neural network.

g. *Qualitative improvements*

The forecast quality of ConvE-STF is analyzed as a function of the weather situations. The RMSE improvement of ConvE-STF and gradient boosting machine models compared to the AROME closest grid point is shown in Fig. 12. The

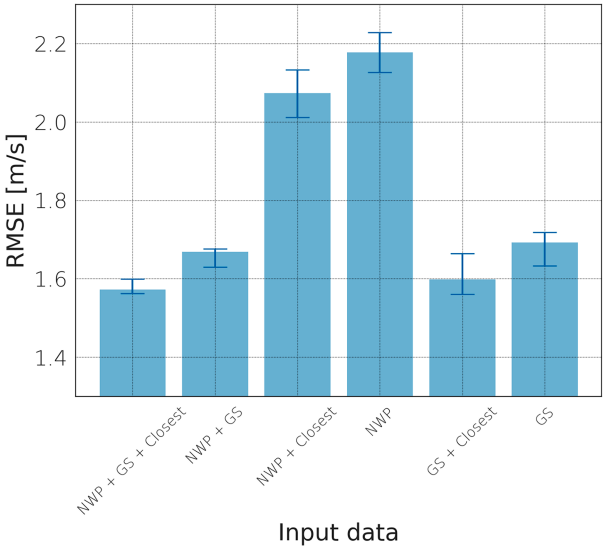

FIG. 11. Sensitivity of the RMSE to the input data. Blue bars show the generalized RMSE, and error bars show the 50% interquantile range over the six splits. Input data are the combination of neighboring GS, NWP, and closest NWP grid point (Closest).

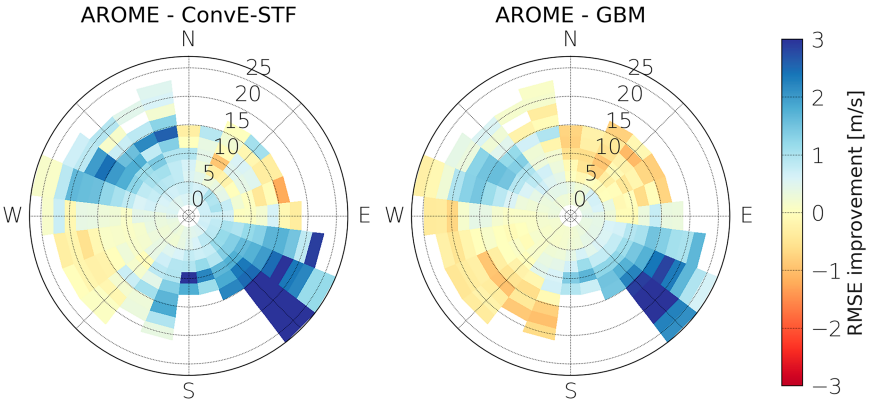

FIG. 12. RMSE improvement between ConvE-STF, GBM, and AROME. The RMSE improvement [RMSE(AROME) − RMSE(model)] is shown in color, with blue sectors indicating a RMSE decrease compared to AROME and red sectors indicating a RMSE increase. The RMSE improvement is plotted as a function of wind direction and wind speed.

ConvE-STF model shows general improvement in RMSE compared to AROME, with a RMSE reduction for most wind speed and direction. It shows the model's skills at postprocessing numerical weather prediction in most weather situations.

The patterns are relatively similar for both models, but ConvE-STF is notably more efficient than the gradient boosting machine for southwest-blowing winds. This can be due to the processing of a larger amount of coastal in situ measurements situated upwind from the target station.

However, both models fail to improve the RMSE for northeast and southwest winds with an increased error compared to the AROME closest grid point. It is important to note that such winds are relatively rare in the eastern Gulf of Lion. Thus, this likely illustrates a shortcoming of the considered training configurations with 2-yr-long datasets. When such wind situations are not present in the training dataset, deep learning models cannot extrapolate during the test phase for so-called out-of-distribution samples.

6. Conclusions and discussion

This paper proposes a deep learning architecture for the probabilistic wind speed forecast at sea. It uses convolutional neural network to process a large amount of input data and is compared to state-of-the-art statistical methods. Several probabilistic assumptions are proposed for multivariate probabilistic forecasting. A Gaussian posterior assumption is compared to normalizing flows and quantile approaches. The proposed method proves skillful at improving the short-term wind forecast (1–6 h ahead) at a target offshore location, with a 40% reduction in RMSE compared to the numerical weather prediction forecast. Other baseline methods improve the forecasts by 14% for analogs to 26% for the gradient boosting machine. It stresses the importance of numerical weather prediction postprocessing for offshore applications. Furthermore, the proposed architecture can emulate probabilistic forecasts with satisfying reliability.

The proposed ConvE-STF architecture shows the best performance in terms of deterministic and probabilistic metrics. It shows an acceptable forecast reliability, with a marginal gain for a Gaussian assumption compared to normalizing flows. Normalizing flow addition can reproduce highly non-Gaussian behaviors for a relatively low computational cost. This can be of great use for multivariate probabilistic metocean forecasting. Other generative models such as GAN, VRAE, or diffusion models could probably achieve similar results and were not explored in this study. Normalizing flows, however, provide a simple yet efficient method to relax the parametric assumption on the posterior distribution.

The use of deep learning methods allows the integration of various sources of data. It permits the use of recent neighboring measurements that have a great impact on the forecast correction. In the context of offshore operations, it shows the opportunity of postprocessing numerical weather prediction using coastal measurements. Moreover, once trained, deep learning models run fast and could enable short-term operational decision-making based on high-frequency forecasts.

Normalizing flows are used as an add-on block to the ConvE-STF architecture with the Gaussian assumption. The normalizing flow transformation conditioning can be constructed in different ways. It is applied in this paper for each lead time independently, and the sampling is to be done for each lead time. The temporal correlation between lead times is not explicit. Whether normalizing flows can be used to jointly model the temporal correlation and variable correlation is still an open question (Dumas et al. 2022).

The considered dataset has inherent limitations. It would be beneficial to complement the study with an extended dataset. The forecast horizon is here limited to 6 h after forecast issue time. In real operational contexts, offshore operation planning and execution (Gintautas and Sørensen 2017) would likely require the extension to 24-h forecasts. Operational NWP forecasts fulfill this requirement (Bauer et al. 2015). Our experiments also assess how the length of the training dataset impacts the forecasting performance of the proposed deep learning scheme. While we retrieve significant average improvement compared with the operational NWP forecast using a 2-yr-long training dataset, we also point out limitations

for rare events, especially southwest and northeast winds in our case study. This is likely a limiting factor for a complete forecast evaluation (Schultz et al. 2021); however, it shows that a skillful data-driven model can be obtained using 2 years of training data. Related studies applied to wind speed forecasting often use shorter or similar datasets to train postprocessing models (Zheng et al. 2022; Gallego et al. 2011; Wang et al. 2017). Extending the considered dataset to longer time series strongly depends on the availability of longer time series of offshore measurements and requires the deployment of dedicated in situ observatories (Marcille et al. 2023). The availability of ensemble NWP forecasts also seems appealing both as a complementary benchmarking baseline as well as to explore how deep learning schemes could benefit from ensemble forecasts as input data (Grönquist et al. 2021). Furthermore, it would be very beneficial to compare the forecasts' reliability with the ensemble prediction of AROME to assess the impact of data representativity on forecast calibration.

Other sources of data could be used to improve the postprocessing of numerical wind forecast. For offshore surface winds, sea surface roughness data through satellite synthetic aperture radar (SAR) images provide high-resolution information (Mouche et al. 2012). To date, SAR images have too low temporal availability (2–3 days) to be integrated into an operational postprocessing model. Further studies on the impact of marine exogenous variables for offshore wind forecasting could be considered.

This study could be extended to jointly forecast wind and wave parameters (Ahmadreza et al. 2008). Potential non-Gaussian distributions are expected between wind and wave parameter forecast uncertainties. From there, the value of the forecast could be evaluated with regard to probabilistic operational decision-making by modeling a realistic maintenance operation (Gintautas and Sørensen 2017; Catterson et al. 2016). The model reliability is then a crucial parameter to justify the operational use of probabilistic forecasts.

Acknowledgments. This research has been supported by France Energies Marines and the French Government, managed by the Agence Nationale de la Recherche under the Investissements d'Avenir program, with the reference ANR-10-IEED-0006-34. This work was carried out in the framework of the FLOWTOM project. It is supported by the ANR project OceaniX.

Data availability statement. Meteorological data used in this study are available online through the MeteoNet dataset. The code developed for the probabilistic short-term forecasting is accessible via https://github.com/rmarcille/conve_stf_meteonet.git (Marcille 2024b) and uses code from Lguensat et al. (2017) for analogs and Durkan et al. (2020) for normalizing flows. The preprocessed dataset is accessible through Marcille (2024a).

REFERENCES

Afrasiabi, M., M. Mohammadi, M. Rastegar, and S. Afrasiabi, 2021: Advanced deep learning approach for probabilistic wind speed forecasting. *IEEE Trans. Ind. Inf.*, **17**, 720–727, https://doi.org/10.1109/TII.2020.3004436.

Ahmadreza, Z., D. Solomatine, A. Azimian, and A. Heemink, 2008: Learning from data for wind–wave forecasting. *Ocean Eng.*, **35**, 953–962, https://doi.org/10.1016/j.oceaneng.2008.03.007.

Akiba, T., S. Sano, T. Yanase, T. Ohta, and M. Koyama, 2019: Optuna: A next-generation hyperparameter optimization framework. *KDD'19: Proc. 25th ACM SIGKDD Int. Conf. on Knowledge Discovery & Data Mining*, Anchorage, AK, Association for Computing Machinery, 2623–2631, https://doi.org/10.1145/3292500.3330701.

Archer, C. L., and Coauthors, 2014: Meteorology for coastal/offshore wind energy in the United States: Recommendations and research needs for the next 10 years. *Bull. Amer. Meteor. Soc.*, **95**, 515–519, https://doi.org/10.1175/BAMS-D-13-00108.1.

Bauer, P., A. Thorpe, and G. Brunet, 2015: The quiet revolution of numerical weather prediction. *Nature*, **525**, 47–55, https://doi.org/10.1038/nature14956.

Bazionis, I. K., and P. S. Georgilakis, 2021: Review of deterministic and probabilistic wind power forecasting: Models, methods, and future research. *Electricity*, **2**, 13–47, https://doi.org/10.3390/electricity2010002.

Bergstra, J., R. Bardenet, Y. Bengio, and B. Kégl, 2011: Algorithms for hyper-parameter optimization. *NIPS'11: Proc. 24th Int. Conf. on Neural Information Processing Systems*, Granada, Spain, Association for Computing Machinery, 2546–2554, https://dl.acm.org/doi/10.5555/2986459.2986743.

Candille, G., and O. Talagrand, 2005: Evaluation of probabilistic prediction systems for a scalar variable. *Quart. J. Roy. Meteor. Soc.*, **131**, 2131–2150, https://doi.org/10.1256/qj.04.71.

Catterson, V. M., D. McMillan, I. Dinwoodie, M. Revie, J. Dowell, J. Quigley, and K. Wilson, 2016: An economic impact metric for evaluating wave height forecasters for offshore wind maintenance access. *Wind Energy*, **19**, 199–212, https://doi.org/10.1002/we.1826.

Det Norske Veritas, 2011: DNV-OS-H101: Marine operations, general. Det Norske Veritas Tech. Rep., 55 pp., https://pdfcoffee.com/dnv-os-h101-marine-operations-general-pdf-free.html.

Dinh, L., J. Sohl-Dickstein, and S. Bengio, 2017: Density estimation using Real NVP. arXiv, 1605.08803v3, https://doi.org/10.48550/arXiv.1605.08803.

Dumas, J., A. Wehenkel, D. Lanaspeze, B. Cornélusse, and A. Sutera, 2022: A deep generative model for probabilistic energy forecasting in power systems: Normalizing flows. *Appl. Energy*, **305**, 117871, https://doi.org/10.1016/j.apenergy.2021.117871.

Durkan, C., A. Bekasov, I. Murray, and G. Papamakarios, 2019: Neural spline flows. *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, H. M. Wallach et al., Eds., Curran Associates, Inc., 7511–7522.

——, ——, ——, and ——, 2020: nflows: Normalizing flows in PyTorch, v0.14. Zenodo, accessed 12 December 2022, https://doi.org/10.5281/zenodo.4296287.

Friedman, J. H., 2001: Greedy function approximation: A gradient boosting machine. *Ann. Stat.*, **29**, 1189–1232, https://doi.org/10.1214/aos/1013203451.

Fritsch, F. N., and R. E. Carlson, 1980: Monotone piecewise cubic interpolation. *SIAM J. Numer. Anal.*, **17**, 238–246, https://doi.org/10.1137/0717021.

Gallego, C., P. Pinson, H. Madsen, A. Costa, and A. Cuerva, 2011: Influence of local wind speed and direction on wind

power dynamics—Application to offshore very short-term forecasting. *Appl. Energy*, **88**, 4087–4096, https://doi.org/10.1016/j.apenergy.2011.04.051.

Gilbert, C., J. Browell, and D. McMillan, 2021: Probabilistic access forecasting for improved offshore operations. *Int. J. Forecasting*, **37**, 134–150, https://doi.org/10.1016/j.ijforecast.2020.03.007.

Gintautas, T., and J. D. Sørensen, 2017: Improved methodology of weather window prediction for offshore operations based on probabilities of operation failure. *J. Mar. Sci. Eng.*, **5**, 20, https://doi.org/10.3390/jmse5020020.

Gneiting, T., and A. E. Raftery, 2007: Strictly proper scoring rules, prediction, and estimation. *J. Amer. Stat. Assoc.*, **102**, 359–378, https://doi.org/10.1198/016214506000001437.

——, L. I. Stanberry, E. P. Grimit, L. Held, and N. A. Johnson, 2008: Assessing probabilistic forecasts of multivariate quantities, with an application to ensemble predictions of surface winds. *Test*, **17**, 211–235, https://doi.org/10.1007/s11749-008-0114-x.

Goodfellow, I., Y. Bengio, and A. Courville, 2016: *Deep Learning*. MIT Press, 780 pp.

Grönquist, P., C. Yao, T. Ben-Nun, N. Dryden, P. Dueben, S. Li, and T. Hoefler, 2021: Deep learning for post-processing ensemble weather forecasts. *Philos. Trans. Roy. Soc.*, **A379**, 20200092, https://doi.org/10.1098/rsta.2020.0092.

He, Y., H. Li, S. Wang, and X. Yao, 2021: Uncertainty analysis of wind power probability density forecasting based on cubic spline interpolation and support vector quantile regression. *Neurocomputing*, **430**, 121–137, https://doi.org/10.1016/j.neucom.2020.10.093.

Higashiyama, K., Y. Fujimoto, and Y. Hayashi, 2018: Feature extraction of NWP data for wind power forecasting using 3D-convolutional neural networks. *Energy Procedia*, **155**, 350–358, https://doi.org/10.1016/j.egypro.2018.11.043.

James, E. P., S. G. Benjamin, and M. Marquis, 2018: Offshore wind speed estimates from a high-resolution rapidly updating numerical weather prediction model forecast dataset. *Wind Energy*, **21**, 264–284, https://doi.org/10.1002/we.2161.

Larvor, G., L. Berthomier, V. Chabot, B. Le Pape, B. Pradel, and L. Perez, 2020: MeteoNet, an open reference weather dataset by Meteo-France. Météo France, accessed 6 June 2022, https://github.com/meteofrance/meteonet.

Leontaris, G., O. Morales-Nápoles, and A. R. M. R. Wolfert, 2016: Probabilistic scheduling of offshore operations using copula based environmental time series—An application for cable installation management for offshore wind farms. *Ocean Eng.*, **125**, 328–341, https://doi.org/10.1016/j.oceaneng.2016.08.029.

Lguensat, R., P. Tandeo, P. Ailliot, M. Pulido, and R. Fablet, 2017: The analog data assimilation. *Mon. Wea. Rev.*, **145**, 4093–4107, https://doi.org/10.1175/MWR-D-16-0441.1.

Marcille, R., 2024a: Conve-stf dataset. Zenodo, accessed 1 March 2024, https://doi.org/10.5281/zenodo.11065946.

——, 2024b: rmarcille/conve_stf_meteonet: v1.1.0. Zenodo, accessed 1 March 2024, https://doi.org/10.5281/zenodo.11066138.

——, M. Thiébaut, P. Tandeo, and J.-F. Filipot, 2023: Gaussian mixture models for the optimal sparse sampling of offshore wind resource. *Wind Energy Sci.*, **8**, 771–786, https://doi.org/10.5194/wes-8-771-2023.

McKinley, S., and M. Levine, 1998: Cubic spline interpolation. *Coll. Redwoods*, **45**, 1049–1060.

Messner, J. W., P. Pinson, J. Browell, M. B. Bjerregård, and I. Schicker, 2020: Evaluation of wind power forecasts—An up-to-date view. *Wind Energy*, **23**, 1461–1481, https://doi.org/10.1002/we.2497.

Mouche, A. A., F. Collard, B. Chapron, K.-F. Dagestad, G. Guitton, J. A. Johannessen, V. Kerbaol, and M. W. Hansen, 2012: On the use of Doppler shift for sea surface wind retrieval from SAR. *IEEE Trans. Geosci. Remote Sens.*, **50**, 2901–2909, https://doi.org/10.1109/TGRS.2011.2174998.

Muschinski, T., G. J. Mayr, T. Simon, N. Umlauf, and A. Zeileis, 2022: Cholesky-based multivariate Gaussian regression. *Econometrics Stat.*, **29**, 261–281, https://doi.org/10.1016/j.ecosta.2022.03.001.

Obakrim, S., V. Monbet, N. Raillard, and P. Ailliot, 2023: Learning the spatiotemporal relationship between wind and significant wave height using deep learning. *Environ. Data Sci.*, **2**, e5, https://doi.org/10.1017/eds.2022.35.

Optis, M., A. Kumler, J. Brodie, and T. Miles, 2021: Quantifying sensitivity in numerical weather prediction-modeled offshore wind speeds through an ensemble modeling approach. *Wind Energy*, **24**, 957–973, https://doi.org/10.1002/we.2611.

Pichugina, Y. L., and Coauthors, 2017: Assessment of NWP forecast models in simulating offshore winds through the lower boundary layer by measurements from a ship-based scanning Doppler Lidar. *Mon. Wea. Rev.*, **145**, 4277–4301, https://doi.org/10.1175/MWR-D-16-0442.1.

Pinson, P., 2012: Adaptive calibration of (u, v)-wind ensemble forecasts. *Quart. J. Roy. Meteor. Soc.*, **138**, 1273–1284, https://doi.org/10.1002/qj.1873.

——, and R. Girard, 2012: Evaluating the quality of scenarios of short-term wind power generation. *Appl. Energy*, **96**, 12–20, https://doi.org/10.1016/j.apenergy.2011.11.004.

Platzer, P., P. Yiou, P. Naveau, J.-F. Filipot, M. Thiébaut, and P. Tandeo, 2021: Probability distributions for analog-to-target distances. *J. Atmos. Sci.*, **78**, 3317–3335, https://doi.org/10.1175/JAS-D-20-0382.1.

Rasul, K., A.-S. Sheikh, I. Schuster, U. Bergmann, and R. Vollgraf, 2021: Multivariate probabilistic time series forecasting via conditioned normalizing flows. arXiv, 2002.06103v3, https://doi.org/10.48550/arXiv.2002.06103.

Rezende, D., and S. Mohamed, 2015: Variational inference with normalizing flows. *Proceedings of the 32nd International Conference on Machine Learning*, Vol. 37, PMLR, 1530–1538, https://proceedings.mlr.press/v37/rezende15.html.

Ruiz, J., P. Ailliot, T. T. T. Chau, P. Le Bras, V. Monbet, F. Sévellec, and P. Tandeo, 2022: Analog data assimilation for the selection of suitable general circulation models. *Geosci. Model Dev.*, **15**, 7203–7220, https://doi.org/10.5194/gmd-15-7203-2022.

Sacco, M. A., J. J. Ruiz, M. Pulido, and P. Tandeo, 2022: Evaluation of machine learning techniques for forecast uncertainty quantification. *Quart. J. Roy. Meteor. Soc.*, **148**, 3470–3490, https://doi.org/10.1002/qj.4362.

Schaeffer, A., P. Garreau, A. Molcard, P. Fraunié, and Y. Seity, 2011: Influence of high-resolution wind forcing on hydrodynamic modeling of the Gulf of Lions. *Ocean Dyn.*, **61**, 1823–1844, https://doi.org/10.1007/s10236-011-0442-3.

Schultz, M. G., C. Betancourt, B. Gong, F. Kleinert, M. Langguth, L. H. Leufen, A. Mozaffari, and S. Stadtler, 2021: Can deep learning beat numerical weather prediction? *Philos. Trans. Roy. Soc.*, **A379**, 20200097, https://doi.org/10.1098/rsta.2020.0097.

Slingo, J., and T. Palmer, 2011: Uncertainty in weather and climate prediction. *Philos. Trans. Roy. Soc.*, **A369**, 4751–4767, https://doi.org/10.1098/rsta.2011.0161.

Sward, J. A., T. R. Ault, and K. M. Zhang, 2023: Spatial biases revealed by LiDAR in a multiphysics WRF ensemble

designed for offshore wind. *Energy*, **262**, 125346, https://doi.org/10.1016/j.energy.2022.125346.

Tambke, J., M. Lange, U. Focken, J.-O. Wolff, and J. A. T. Bye, 2005: Forecasting offshore wind speeds above the North Sea. *Wind Energy*, **8**, 3–16, https://doi.org/10.1002/we.140.

Taylor, J. W., and J. Jeon, 2018: Probabilistic forecasting of wave height for offshore wind turbine maintenance. *Eur. J. Oper. Res.*, **267**, 877–890, https://doi.org/10.1016/j.ejor.2017.12.021.

Vannitsem, S., and Coauthors, 2021: Statistical postprocessing for weather forecasts: Review, challenges, and avenues in a big data world. *Bull. Amer. Meteor. Soc.*, **102**, E681–E699, https://doi.org/10.1175/BAMS-D-19-0308.1.

Wang, H.-z., G.-q. Li, G.-b. Wang, J.-c. Peng, H. Jiang, and Y.-t. Liu, 2017: Deep learning based ensemble approach for probabilistic wind power forecasting. *Appl. Energy*, **188**, 56–70, https://doi.org/10.1016/j.apenergy.2016.11.111.

Zheng, Z., L. Wang, L. Yang, and Z. Zhang, 2022: Generative probabilistic wind speed forecasting: A variational recurrent autoencoder based method. *IEEE Trans. Power Syst.*, **37**, 1386–1398, https://doi.org/10.1109/TPWRS.2021.3105101.

Zou, R., M. Song, Y. Wang, J. Wang, K. Yang, and M. Affenzeller, 2022: Deep non-crossing probabilistic wind speed forecasting with multi-scale features. *Energy Convers. Manage.*, **257**, 115433, https://doi.org/10.1016/j.enconman.2022.115433.