Continuous and Distribution-free Probabilistic Wind Power Forecasting

Honglin Wen, *Sudent Member, IEEE,* Pierre Pinson, *Fellow, IEEE,* Jinghuan Ma, Jie Gu, and Zhijian Jin

***Abstract*—We present a data-driven approach for probabilistic wind power forecasting based on conditional normalizing flow (CNF). In contrast with the existing, this approach is distribution-free (as for non-parametric and quantile-based approaches) and can directly yield continuous probability densities, hence avoiding quantile crossing. It relies on a base distribution and a set of bijective mappings. Both the shape parameters of the base distribution and the bijective mappings are approximated with neural networks. Spline-based conditional normalizing flow is considered owing to its universal approximation capability. Over the training phase, the model sequentially maps input examples onto samples of base distribution, where parameters are estimated through maximum likelihood. To issue probabilistic forecasts, one eventually map samples of the base distribution into samples of a desired distribution. Case studies based on open datasets validate the effectiveness of the proposed model, and allows us to discuss its advantages and caveats with respect to the state of the art.**

***Index Terms*—Conditional normalizing flow, deep learning, density estimation, probabilistic forecasting, wind power.**

I. Introduction

A. Motivation

As an essential tool to assess and accommodate wind power generation uncertainty, probabilistic wind power forecasting (PWPF) has gained increasing interest in recent decades. It generally takes numerical weather prediction and historical values as input features, in order to model and communicate the probability density of wind power generation at some time in the future. Such densities may be for a unique lead time and location (hence, univariate), or jointly for several lead times and/or locations (referred to as multivariate) [1].

A classical approach to both univariate and multivariate probabilistic forecasting relies on assumptions for the distribution of future wind power generation, the parameters of which are estimated via statistical and machine learning methods. For instance, the Gaussian distribution assumption could be used in univariate probabilistic forecasting, and the multivariate Gaussian assumption can be adopted in multivariate probabilistic forecasting [2]. Many more assumptions can be considered, e.g., Beta, Generalized Logit-Normal, etc. Although it is convenient to develop models based on such assumptions, the distribution of wind power at hand may not match the assumptions. This is primarily due to the wind power generation process, in other words, the nonlinear power curve that converts energy from the wind into electric power [3]. Concretely, the characteristics of wind power generation distributions differ a lot depending on predicted weather conditions, as illustrated by [4] for instance. This has motivated many to look for distribution-free approaches, i.e., that do not rely on a specific assumption for the densities to model and communicate as forecasts.

Certainly the most popular distribution-free approach, also referred to as non-parametric, is quantile regression (QR) [5], which allows to relax the use of distributional assumptions for the case of univariate probabilistic forecasting. It has achieved great success in the Global Energy Forecasting Competition 2014 (GEFCom 2014) for instance, and has become a mainstream solution owing to its performance and simplicity of use. However, it requires parallel models to be fitted for each quantile, which raises the cost of computation when the whole distribution is needed. In addition, it only provides discrete quantiles, which may lead to quantile crossing – quantiles of the whole distribution are inconsistent. As for the multivariate probabilistic forecasting, it has become common now to decouple the estimation of the marginal probability density function (PDF) of each variable and of the interdependence structure [6]. Complex interdependence structures can be modeled using copula models [7]. By estimating the marginal PDF via non-parametric methods and modeling the complex interdependence structure, the copula method relaxes the commonly used multivariate Gaussian distribution assumption. However, the copula-based approach relies on strong assumptions regarding the probabilistic calibration of predicted marginals, while it often underestimate the strength of the dependence structure among the various variables. Eventually, it remains an open issue to develop an efficient, continuous and distribution-free probabilistic forecasting model that obtains whole distribution at once.

B. Related Works

Univariate probabilistic forecasting usually translates to communicating quantile forecasts, prediction intervals (PI), and predictive densities. The quantile forecasts and PI are specific characteristics of predictive densities, which are most often obtained by QR. Based on this approach, several machine learning models such as neural network (NN) [8] and gradient boost machine [9] have been adopted to estimate the conditional quantile function. It is then simple and effective to construct PI with two corresponding quantile functions. A $(1-\beta)\times 100\%$ PI can be constructed by the pair of quantiles $(\alpha, 1-\beta+\alpha)$ where $\alpha \in (0, \beta)$, for instance $(\beta/2, 1-\beta/2)$ as typically derived in the literature [10]. However, both PIs and quantiles only provide partial information of probability densities, the applications of which can hardly cover stochastic power system operations where amounts of samples are often required.

Honglin Wen, Jinghuan Ma, Jie Gu, and Zhijian Jin are with Department of Electrical Engineering, Shanghai Jiao Tong University.
Honglin Wen, Pierre Pinson are with the Technical University of Denmark, Department of Technology, Management and Economics.

As a result of this, it has been an active research topic to communicate densities in the PWPF community. Besides the aforementioned method with distribution assumptions (often referred to as parametric models), resampling and advanced density estimation techniques have been adopted, as reviewed in [1], [11]. The idea of resampling method lies in estimating the PDF of empirical errors of point forecasts, which makes the method naturally distribution-free. In order to issue conditional densities for the PWPF, fuzzy inference has been applied to classify the forecast conditions into specific amounts of modes [4]. But such finite classifications cannot continuously adapt to all forecasting conditions. Furthermore, the quality of the estimated densities is strongly related to the performance of utilized point-forecast model. The non-parametric density estimation method, namely kernel density estimation (KDE) has been popular among the PWPF community due to its universal approximation capability. In particular, it generally deduces the density of a finite population selected by k-nearest neighbors [12]. As with the resampling method, this method is still limited in modeling conditional densities, since the employed k-nearest neighbors operation is restricted in dealing with heterogeneous distribution. That said, once k is fixed, the KDE-based model cannot adaptively select the finite population. In addition, the k-nearest neighbor operation suffers from the curse of dimension. Recently, mixture density network has been applied in PWPF, as it can model more complex distribution through the comic combination of Gaussian distribution [13]. But it would get stuck in mode collapse issues, which translates to saying that the ultimate distribution would collapse into a Gaussian distribution [14].

Multivariate probabilistic forecasting often communicates *scenarios* that are samples drawn from predictive densities. The scenario generation procedure is based on probability integral transform (PIT) and the interdependence structure characterized by a covariance matrix [6]. Concretely, one draws realizations from the estimated multivariate standard Gaussian distribution, and converts the realizations into scenarios of wind power generation via inverse PIT. Besides, the emerging approach is to directly learn multivariate densities based on advanced generative models such as the generative adversarial network (GAN) adopted in [15]. The GAN is composed of a generator and a discriminator, where the generator is responsible for generating scenarios in the operation stage. Although it is computationally more efficient than the copula approach, it suffers from notorious training instability caused by the game between the generator and discriminator in training phase [16]. The most related work is [17], which compares the performance of several generative models, i.e., GAN, variational auto-encoder, and normalizing flow (NF). But their primary focus is to compare the performances of deep learning based generative model, therefore leaving issues such as applicability of NF and the relationship between NF with existing models uncovered. Our work goes beyond offering a complete distribution-free forecasting framework, and it distinguishes itself from existing works by uncovering its relationship with several commonly used methods in the PWPF community.

C. Proposed Method and Main Contributions

As a basis for this work, we get inspiration from [4] and [18], which relied on the idea of transforming samples of bounded stochastic process at hand to make them more suitable to be modeled by a Gaussian (or multivariate Gaussian) variable. Indeed, it is allowed by the *conservation of probability measure* [19], which translates into saying that one can transform a variable that follows an arbitrary kind of distribution into a variable that follows a desired distribution with the assistance of bijective mapping (transform). Here, instead of using a manually designed transform, we implement such transforms via the NF [20], [21]. An NF framework is composed of a base distribution and a sequence of trainable bijective mappings. Both the shape parameters of base distribution and bijective mappings are modeled by neural networks (NNs). Besides, such transforms ought to be non-affine so that the model can flexibly characterize the wind power distribution under different conditions. Concretely, we establish a distribution-free PWPF model based on a conditional auto-regressive NF [22], which is applicable to both univariate PWPF and multivariate PWPF applications. Unlike copula models where the marginal PDF and interdependence structure are modeled separately, here the joint probability density is derived through the chain rule of probability, i.e., the product of conditional probability densities. In particular, such conditional probability densities are also dependent on input features [23]. Spline-based NF [24] is adopted to acquire the power of universal distribution approximation. All the parameters are estimated simultaneously based on the maximum likelihood. Case studies validate the effectiveness of the proposed model, which achieves state-of-the-art.

The main contributions of the paper are: (i) The proposal of a distribution-free PWPF model, which models the whole predictive distribution, and suffices to handle the bounded characteristics of wind power. (ii) The demonstration of its applicability to both univariate PWPF and multivariate PWPF problems, which avoids quantile crossing issue in the univariate PWPF and efficiently models the interdependence structure in the multivariate PWPF. (iii) A new perspective for conditional PDF estimation for PWPF based on the function theory, which offers complimentary understanding to merits and caveats of distribution-free approaches versus parametric approaches.

The remainder of this paper is organized as follows. In section II, the methodological components of normalizing flows are introduced. Our approach to their application to univariate and multivariate wind power probabilistic forecasting is described in section III. Section IV summarizes data sources and experiment implementation. The results obtained are presented in Section V, and the performance comparison with existing models is discussed. Section VI concludes this paper.

II. Methodological Components

Denote the input features at time t as $\boldsymbol{x}_t \in \mathbb{R}^D$ and the predictive wind power with look-ahead time Δ as $\boldsymbol{y}_{t+\Delta} \in \mathbb{R}^d$, where D and d denote the dimensions of variables.

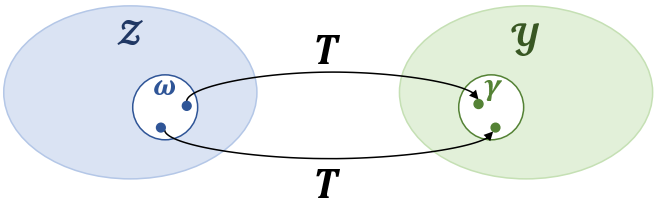

Fig. 1: Illustration of transform

The essence of PWPF is to estimate the conditional PDF $p(\boldsymbol{y}_{t+\Delta}|\boldsymbol{x}_t)$. However, such estimation is nontrivial as $\boldsymbol{x}_t$ is a continuous variable. In this paper, we address this issue by means of CNF, whose base distribution and bijective mappings are parameterized by input features. In the remaining parts, we introduce preliminaries i.e., conservation of probability measure and describes the framework of CNF. Consequently, we discuss the relationship between this method and classical approaches. Without loss of generality, we use the notation of multivariate here.

A. Preliminaries

The most important base properties to consider for normalizing flows are the concepts of conservation of probability measure and diffeomorphism.

Definition 1 (Conservation of Probability Measure) : Denote the PDF defined on set $\mathcal{Z}$ as $p_z : \mathcal{Z} \to [0, +\infty)$, the PDF defined on set $\mathcal{Y}$ as $p_y : \mathcal{Y} \to [0, +\infty)$, and the transform as $T : \mathcal{Z} \to \mathcal{Y}$. For any subset $\omega \subseteq \mathcal{Z}$, we have

$$\int_{\boldsymbol{z}\in\omega} p_z(\boldsymbol{z})d\mu(\boldsymbol{z}) = \int_{\boldsymbol{y}\in\gamma} p_y(\boldsymbol{y})d\nu(\boldsymbol{y}), \tag{1}$$

where $\gamma = \{T(\boldsymbol{z})|\boldsymbol{z} \in \omega\}$, $d\mu(\boldsymbol{z})$ and $d\nu(\boldsymbol{y})$ are the integration measures [19].

Definition 2 (Diffeomorphism) A diffeomorphism is an invertible mapping that maps one differentiable manifold to another s.t. both the function and its inverse are smooth [25].

The NF model is an application of conservation of probability measure by specifying T as diffeomorphism and the two sets $\mathcal{Z}$ along with $\mathcal{Y}$ as Euclidean space [21]. Hence, the formula is expressed as

$$\int_{\boldsymbol{z}\in\omega} p_z(\boldsymbol{z})d\boldsymbol{z} = \int_{\boldsymbol{y}\in\gamma} p_y(\boldsymbol{y})d\boldsymbol{y}. \tag{2}$$

By utilizing the change of variable, $\boldsymbol{z} = T^{-1}(\boldsymbol{y})$, we convert the formula into

$$\int_{\boldsymbol{y}\in\gamma} p_z(T^{-1}(\boldsymbol{y}))|\det J_{T^{-1}}(\boldsymbol{y})|d\boldsymbol{y} = \int_{\boldsymbol{y}\in\gamma} p_y(\boldsymbol{y})d\boldsymbol{y}, \tag{3}$$

where $J_{T^{-1}}(\boldsymbol{y})$ denotes the Jacobian matrix s.t.

$$J_{T^{-1}}(\boldsymbol{y})_{i.j} = \frac{\partial y_i}{\partial z_j}. \tag{4}$$

As it holds for any subset $\gamma \subseteq \mathcal{Y}$, we have

$$p_y(\boldsymbol{y}) = p_z(T^{-1}(\boldsymbol{y}))|\det J_{T^{-1}}(\boldsymbol{y})|. \tag{5}$$

B. Flow Model for Forecasting

With the introduced conservation of probability measure, we can establish conditional NF models.

1) Flow Framework: Generally, the transform T in an NF consists of a series of diffeomorphisms $T_1, T_2, \ldots, T_K$ [21], i.e.,

$$T = T_1 \circ T_2 \circ \cdots \circ T_K, \tag{6}$$

where $\circ$ denotes the symbol of composition. For each T_k, we denote its input variable as $\boldsymbol{z}^{(k-1)}$ and its output variable as $\boldsymbol{z}^{(k)}$. Particularly, $\boldsymbol{z}^{(0)}$ follows the base distribution $\Phi(\cdot|\boldsymbol{x}_t)$ that is specified by $\boldsymbol{x}_t$, whereas $\boldsymbol{z}^{(K)}$ is equal to output variable $\boldsymbol{y}_{t+\Delta}$.

Two significant calculation passes in NF models are forward pass and inverse pass. Such computations between $\boldsymbol{z}^{(k)}$ and $\boldsymbol{z}^{(k-1)}$ for instance are respectively described as

$$\boldsymbol{z}^{(k)} = T_k(\boldsymbol{z}^{(k-1)}), \tag{7}$$

$$\boldsymbol{z}^{(k-1)} = T_k^{-1}(\boldsymbol{z}^{(k)}). \tag{8}$$

With the sequential transforms, we have

$$\boldsymbol{y} = T(\boldsymbol{z}^{(0)}), \tag{9}$$

$$\boldsymbol{z}^{(0)} = T^{-1}(\boldsymbol{y}). \tag{10}$$

And the Jacobian determinant is computed by

$$\begin{aligned}\log|\det J_T(\boldsymbol{z}^{(0)})| &= \log|\prod_{k=1}^{K}\det J_{T_k}(\boldsymbol{z}^{(k-1)})| \\ &= \sum_{k=1}^{K}\log|\det J_{T_k}(\boldsymbol{z}^{(k-1)})|.\end{aligned} \tag{11}$$

Ultimately, we build the connection between the PDF of $\boldsymbol{z}_0$ and that of $\boldsymbol{y}$, i.e.,

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{z}^{(0)}) + \sum_{k=1}^{K}\log|\det J_{T_k}(\boldsymbol{z}^{(k-1)})|. \tag{12}$$

Such T_k in the NF model is implemented via NNs parameterized by $\boldsymbol{\phi}_k$, and is required to be invertible and have a tractable Jacobian determinant.

2) Conditional Normalizing Flow: In particular, we obtain $p(\boldsymbol{z}^{(k)}|\boldsymbol{x}_t)$ through $p(\boldsymbol{z}^{(k-1)}|\boldsymbol{x}_t)$ and the mapping T_k, which is bijective in $\boldsymbol{z}^{(k-1)}$ as well as $\boldsymbol{z}^{(k)}$ and parameterized by $\boldsymbol{x}_t$. We have

$$\begin{aligned}p(\boldsymbol{z}^{(k)}|\boldsymbol{x}_t) &= p(\boldsymbol{z}^{(k-1)}|\boldsymbol{x}_t)|\frac{\partial \boldsymbol{z}^{(k-1)}}{\partial \boldsymbol{z}^{(k)}}| \\ &= p(T_k^{-1}(\boldsymbol{z}^{(k)}, \boldsymbol{x}_t)|\boldsymbol{x}_t)|\frac{\partial T_k^{-1}(\boldsymbol{z}^{(k)}, \boldsymbol{x}_t)}{\partial \boldsymbol{z}^{(k)}}| \\ &= p(T_k^{-1}(\boldsymbol{z}^{(k)}, \boldsymbol{x}_t)|\boldsymbol{x})|\det J_{T_k}(\boldsymbol{z}^{(k-1)})|.\end{aligned} \tag{13}$$

Consequently, the forward pass and inverse pass in CNF are expressed as

$$\boldsymbol{z}^{(k)} = T_k(\boldsymbol{z}^{(k-1)}; \boldsymbol{x}_t), \tag{14}$$

$$\boldsymbol{z}^{(k-1)} = T_k^{-1}(\boldsymbol{z}^{(k)}; \boldsymbol{x}_t). \tag{15}$$

And the parameters of base distribution Φ are determined by a function of $\boldsymbol{x}_t$, $g(\boldsymbol{x}_t)$. Setting the base distribution as a Gaussian distribution for instance, its parameters $\boldsymbol{\mu}, \Sigma$ are determined as

$$\boldsymbol{\mu}, \Sigma = g(\boldsymbol{x}_t). \tag{16}$$

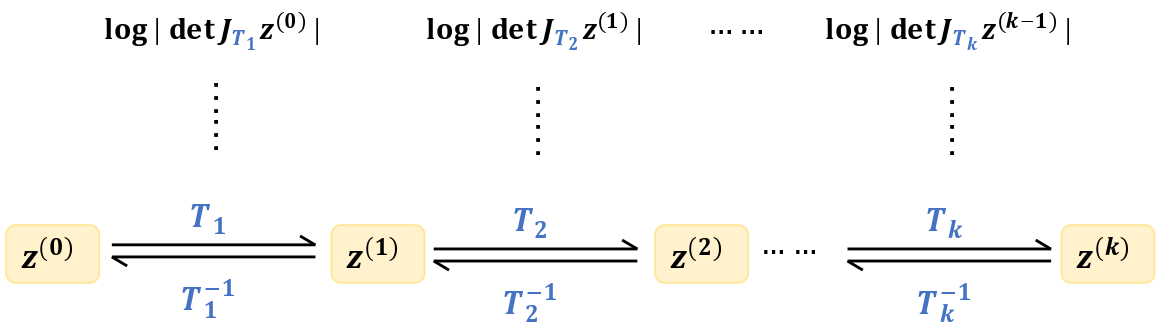

Fig. 2: Framework of flow model

3) Training and Forecasting Procedure: The introduced CNF model is trained based on the maximum likelihood. Denote the training dataset outputs as $Y = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_N\}$ corresponding to inputs $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$. The loss function is defined as

$$\begin{aligned} L &= -\frac{1}{N}\sum_{n=1}^{N}\log p(\boldsymbol{y}_n|\boldsymbol{x}_n) \\ &= -\frac{1}{N}\sum_{n=1}^{N}[\log p(T^{-1}(\boldsymbol{y}_n, \boldsymbol{x}_n)) + \log|\det J_T(T^{-1}(\boldsymbol{y}_n, \boldsymbol{x}_n))|]. \end{aligned} \tag{17}$$

In the training stage we estimate $\{\phi_k, \boldsymbol{\mu}, \Sigma | k = 1, 2, \cdots, K\}$ via the inverse pass. To issue forecasts, we feed $\boldsymbol{x}_t^*$ into the base distribution and all transforms, and derive the PDF via the forward pass.

C. Relationship with Classical Methods

The classical methods can be explained by this framework. Here we consider the base distribution of these methods as Gaussian distribution, i.e., $\boldsymbol{z}^{(0)} \sim \mathcal{N}(\cdot|\boldsymbol{x}_t)$.

1) Gaussian Framework: Models with the assumption of Gaussian [26], [27] can be translated into setting the transform of NF models as an affine transform. That is,

$$\boldsymbol{y}_{t+\Delta} = T(\boldsymbol{z}^{(0)}) = \boldsymbol{A}\boldsymbol{z}^{(0)} + \boldsymbol{b}, \tag{18}$$

where $\boldsymbol{A}$ and $\boldsymbol{b}$ are the corresponding matrix and vector. As the base distribution is a Gaussian distribution, $\boldsymbol{y}_{t+\Delta}$ still obeys Gaussian distribution after a set of affine transforms.

2) Logit-Normal Transform: The logit-normal transform approach [18] can be interpreted as setting the transform T as a sigmoid function, which operates element-wise, i.e.,

$$\boldsymbol{y}_{t+\Delta,i} = \frac{\exp(\boldsymbol{z}_i^{(0)})}{1 + \exp(\boldsymbol{z}_i^{(0)})}. \tag{19}$$

3) Mixture Density Network: The model based on mixture density network [13] can be considered as setting T as the conic combination of affine transforms. Denote these component affine transforms as $T_1, T_2, \cdots, T_m$. The transform defined by this model operates as

$$\boldsymbol{y}_{t+\Delta} = T(\boldsymbol{z}^{(0)}) = \sum_{i=1}^{m}\omega_i T_i(\boldsymbol{z}^{(0)}), \tag{20}$$

where ω_i represents the weight.

4) Gaussian Copula: The model based on Gaussian Copula [6] is an instance of NF, which is specified by an element-wise monotone function g and a correlation matrix Σ . That is,

$$\boldsymbol{z}^{(0)} \sim \mathcal{N}(\boldsymbol{0}, \Sigma), \tag{21}$$

$$\boldsymbol{y}_{t+\Delta} = T(\boldsymbol{z}^{(0)}), \tag{22}$$

where T represents the element-wise monotone function g.

Indeed, any desired distribution can be obtained by transforming Gaussian distribution through a specific mapping. Such mapping proceeds each value in the domain in the same manner. This implies that characteristics of the derived wind power distributions remain the same for different wind conditions. Although the conic combination enables deriving more complex distributions compared to Gaussian distributions, it still handles each condition indifferently. With regard to the Gaussian copula model, it is developed for multivariate modeling. By sufficiently modeling the marginal PDF and correlation structure, one can yield the the ultimate joint probability density in a distribution-free approach. However, as mentioned above, it highly relies on the estimation of marginals and tends to underestimate the covariance structure, which often impedes its performance.

III. Forecasting Applications

The basic approach for conditional normalizing flows described in the above can readily be used for forecasting applications, in both univariate and multivariate settings. We choose the Gaussian distribution as base distribution and adopt a non-affine flow to obtain a piece-wise non-Gaussian distribution.

A. Univariate Probabilistic Forecasting

In the univariate case, each intermediary variable z_k and shape parameters of the Gaussian distribution μ, σ are scalars. The shape parameters of base distribution μ, σ are derived via Eq. (16), whereas T_k is a univariate function that operates as

$$z^{(k)} = T_k(z^{(k-1)}; \boldsymbol{x}_t). \tag{23}$$

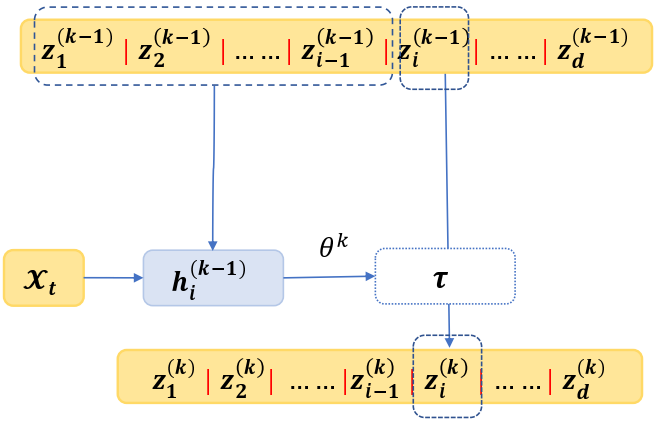

Fig. 3: Illustration of auto-regressive flow

B. Multivariate Probabilistic Forecasting

With the chain rule of probability, we decompose the joint probability density $p(\boldsymbol{z}^{(k)})$ into a product of conditional probability densities, i.e.,

$$p(\boldsymbol{z}^{(k)}) = \prod_{i=1}^{d} p(z_i^{(k)} | \boldsymbol{z}_{1:i-1}^{(k)}). \tag{24}$$

In each conditional density, previous variables $\boldsymbol{z}_{1:i-1}^{(k)}$ serve as conditionals. Specifically, $p(\boldsymbol{z}^{(k)}|\boldsymbol{x}_t)$ is expressed as

$$p(\boldsymbol{z}^{(k)}|\boldsymbol{x}) = \prod_{i=1}^{d} p(z_i^{(k)} | \boldsymbol{z}_{1:i-1}^{(k)}, \boldsymbol{x}_t). \tag{25}$$

Here, we denote the operation in the i-th dimension of T_k as $\tau_i^{(k)}$, which transforms the i-th dimension of of $\boldsymbol{z}^{(k-1)}$, i.e., $z_i^{(k-1)}$ into $z_i^{(k)}$. Thus, the parameters of mapping $\tau_i^{(k)}$, denoted as $\boldsymbol{\theta}_i^{(k)}$, are determined by $\boldsymbol{z}_{1:i-1}^{(k-1)}$ and $\boldsymbol{x}_t$. They are calculated by function $h_i^{(k)}$ realized by an NN. That is

$$\boldsymbol{\theta}_i^{(k)} = h_i^{(k)}(\boldsymbol{z}_{1:i-1}^{(k-1)}, \boldsymbol{x}_t). \tag{26}$$

Hence, $\tau_i^{(k)}$ operates as:

$$z_i^{(k)} = \tau_i^{(k)}(z_i^{(k-1)}; \boldsymbol{\theta}_i^{(k)}), \tag{27}$$

Illustration of such calculation procedure is shown in Fig. 3.

C. Non-affine Transform

The main idea ofspline-based NF is to implement the transform as monotonic spline [24]. Each $\tau_i^{(k)}(z_i^{(k-1)}; \boldsymbol{\theta}_i^{(k)})$ (for univariate case, $\tau_i^{(k)}$ is T_k) is represented as a piecewise function which contains M segments specified by $M+1$ coordinates (knots). The knots are defined by input locations $z_{i0}^{(k-1)}, \cdots, z_{iM}^{(k-1)}$ and corresponding outputs $z_{i0}^{(k)}, \cdots, z_{iM}^{(k)}$, which we denote as $\{(\alpha^{(m)}, \beta^{(m)})|m = 0, \cdots, M\}$ for simplicity. In each interval $[\alpha^{(m-1)}, \alpha^{(m)}]$, $\tau_i^{(k)}$ is a simple monotonic function. And the segments meet at internal knots $\{(\alpha^{(m)}, \beta^{(m)})|m = 1, \cdots, M-1\}$. Specifically, we use monotonic rational-quadratic splines, which are defined by derivatives at internal knots $\{\delta^{(m)}|m = 1, \cdots, M-1\}$ besides the knots. We define

$$s^{(m)} = \frac{\beta^{(m)} - \beta^{(m-1)}}{\alpha^{(m)} - \alpha^{(m-1)}}, \tag{28}$$

$$\xi(z_i^{(k-1)}) = \frac{z_i^{(k-1)} - \alpha^{(m-1)}}{\alpha^{(m)} - \alpha^{(m-1)}}. \tag{29}$$

The rational-quadratic function in the m-th bin is expressed as

$$r_m(\xi) = \beta^{(m-1)} + \frac{(\beta^{(m)} - \beta^{(m-1)})[s^{(m)}\xi^2 + \delta^{(m-1)}\xi(1-\xi)]}{s^{(m)} + [\delta^{(m)} + \delta^{(m-1)} - 2s^{(m)}]\xi(1-\xi)}. \tag{30}$$

That is,

$$\tau_i^{(k)}(z_i^{(k-1)}) = r_m(\xi), \text{if} z_i^{(k-1)} \in [\alpha^{(m-1)}, \alpha^{(m)}]. \tag{31}$$

Specifically, when $z_i^{(k-1)} < \alpha^{(0)}$ or $z_i^{(k-1)} > \alpha^{(M)}$, we set $\tau_i^{(k)}$ as equivalent transform, i.e., $\tau_i^{(k)}(z_i^{(k-1)}) = z_i^{(k-1)}$. As $\tau_i^{(k)}$ is monotonic, the inverse path can be computed analytically by solving a quadratic equation, i.e.,

$$\xi(z_i^{(k-1)}) = \frac{2c}{-b - \sqrt{b^2 - 4ac}}, \tag{32}$$

where

$$\begin{aligned} a =& (\beta^{(m)} - \beta^{(m-1)})(s^{(m)} - \delta^{(m-1)}) \\ &+ (z_i^{(k)} - \beta^{(m-1)})(\delta^{(m)} + \delta^{(m-1)} - 2s^{(m)}), \end{aligned} \tag{33}$$

$$\begin{aligned} b =& (\beta^{(m)} - \beta^{(m-1)})\delta^{(m-1)} \\ &- (z_i^{(k)} - \beta^{(m-1)})(\delta^{(m)} + \delta^{(m-1)} - 2s^{(m)}), \end{aligned} \tag{34}$$

$$c = -s^{(m)}(z_i^{(k)} - \beta^{(m-1)}). \tag{35}$$

All the parameters are obtained from $\boldsymbol{\theta}_i^{(k)}$. By using the spline-based NF, we come to make the transform non-affine.

IV. Case Study

In this paper, we validate the proposed approach in both univariate cases (Case 1, Case 2) and multivariate cases (Case 3, Case 4). Their settings are described as follows and summarized in Table I.

1) Case 1: It is a day-ahead PWPF case based on the GEFCom 2014 data[1], where numerical weather predictions (NWPs) are taken as inputs and predictive PDF of wind power at each timestamp is output.
2) Case 2: It is a very-short-term PWPF case where historical values of wind power are taken as inputs, and predictive PDF of wind power at a future timestamp is output. one-step forecasting is issued here for validation based on the NREL[2] and French wind farm data[3].
3) Case 3: It is a scenario generation case based on the French wind farm data, which considers temporal interdependence. Specifically, we generate scenarios at future six timestamps, which can be used in electricity market.

[1]Available at http://blog.drhongtao.com/2017/03/gefcom2014-load-forecasting-data.html
[2]Available at https://www.nrel.gov/grid/wind-toolkit.html
[3]Available at https://opendata-renewables.engie.com/explore/index

TABLE I: Case study settings

	Type of variable	Input feature	Forecasting horizon	Type of interdependence	Dataset
Case 1	univariate	NWP	24-h	none	GEFCom 2014
Case 2	univariate	historical values	one-step	none	NREL, French wind farm
Case 3	Multivariate	historical values	multi-step	temporal interdependence	French wind farm
Case 4	Multivariate	historical values	one-step	spatial interdependence	NREL

TABLE II: Dataset description

Dataset	Description	Resolution	Samples
GEFCom 2014	NWP that contains wind speed and direction at two altitudes respectively, as well as corresponding wind power values	1-h	16800
French wind farm	Time series of wind power	10-min	52355
NREL	Time series of wind power	15-min	35040

4) Case 4: It is a scenario generation case based on the NREL data, which considers spatial interdependence of multiple sites. Specifically, we choose data of 5 nearby wind farms for validation.

In Case 2, Case 3, and Case 4, the length of input features is determined by preliminary test, which is varied from 4 to 24 and empirically set as 6.

A. Dataset Description

Three open datasets are used for validation, i.e., data from GEFCom 2014, NREL, and French wind farm. The GEFCom 2014 dataset provides NWPs that contain wind speeds and directions at 10-m and 100-m, and corresponding normalized wind power generation. It is an hourly data set collected in 2012 and 2013, and contains a total of 16,800 samples. We randomly select data from 5 wind farms for experiments. The French wind farm data and the NREL data are time series. Data from the French wind farm are collected from four wind turbines, whereas NREL data are generated by simulation at various sites. The resolution of the French wind farm data is 10-min, whereas that of the NREL data is 15-min. Specifically, we select French wind farm data collected in 2013 which contain 52355 samples, and NREL data collected in 2012 which contain 35040 samples for validation. In each case, we split 70% of the data as a training set, 10% as a validation set and 20% as a test set. The information about datasets is summarized in Table II.

B. Assessment Metrics

In this paper, the quality of predictive probability density in univariate cases is assessed by continuous ranked probability score (CRPS) as suggested by [28]. And the quality of predictive probability density in multivariate cases is assessed by scenarios in terms of energy score (ES) and variogram score (VS) as suggested by [29], [30]. All of them are averaged over the whole test data.

1) CRPS: Let $F(\hat{y})$ denote the CDF of predicted wind power $\hat{y}$. The CRPS is defined as:

$$\mathrm{CRPS}(F, y) = \int_{-\infty}^{\infty} (F(\hat{y}) - \mathbb{1}(\hat{y} - y))^2 d\hat{y}, \tag{36}$$

where $\mathbb{1}(\cdot)$ is unit step function, which represents the empirical CDF of observation.

2) ES: Given a set of scenarios $\{\hat{\boldsymbol{y}}^{(i)} | i = 1, \cdots, S\}$ and observations $\boldsymbol{y}$, the ES is defined as

$$\mathrm{ES} = \frac{1}{S}\sum_{i=1}^{S} \|\boldsymbol{y} - \hat{\boldsymbol{y}}^{(i)}\|_2 - \frac{1}{2S^2}\sum_{i=1}^{S}\sum_{j=1}^{S} \|\hat{\boldsymbol{y}}^{(i)} - \hat{\boldsymbol{y}}^{(j)}\|_2, \tag{37}$$

where $\|\cdot\|_2$ is the d-dimensional Euclidean norm.

3) VS: Let y_i and $\hat{y}_i$ respectively denote the i-th dimension of the observation $\boldsymbol{y}$ and a scenario $\hat{\boldsymbol{y}}$. The VS is defined as

$$\mathrm{VS} = \sum_{i,j=1}^{d} (|y_i - y_j|^p - \mathbb{E}(|\hat{y}_i - \hat{y}_j|^p))^2, \tag{38}$$

where

$$\mathbb{E}(|\hat{y}_i - \hat{y}_j|^p) \approx \frac{1}{S}\sum_{s=1}^{S} |\hat{y}_i^{(s)} - \hat{y}_j^{(s)}|^p. \tag{39}$$

Here we set p as 0.5 as suggested by [30].

C. Benchmarks

1) Univariate Cases: We set both parametric models and non-parametric models as benchmarks. For the parametric approaches, we choose NN models that rely on assumptions of Gaussian and logit-normal distributions, and refer to them as NN-G and NN-L respectively. They share the same basic NN structure with the proposed model. As for the non-parametric approaches, we include two popular distribution-free models KDE [12] and quantile regression gradient boosting machine (QRGBM) [9] as benchmarks since they are proved effective in the GEFCom 2014. The QRGBM is an ensemble model that iteratively fits new tree model to minimize the quantile loss. Concretely, in the KDE, we determine the nearest 100 neighbors of each test sample and use their corresponding wind power values to estimate the predictive PDF. In addition, the climatology model is adopted as a naive benchmark model, which estimates the predictive probability density using all training data.

TABLE III: CRPS on GEFCom 2014 (percentage of nominal capacity)

	1	3	5	7	9
Climatology	19.30	18.38	21.36	18.10	18.79
NN-G	9.45	8.98	8.51	7.43	8.48
NN-L	9.33	8.62	8.88	7.40	8.88
QRGBM	9.72	8.57	8.32	7.62	8.27
KDE	10.07	8.76	8.64	7.76	8.56
Proposed model	9.22	8.35	8.14	7.09	8.28

2) Multivariate Cases: For multivariate cases, we mainly use NN-G, and NN-L as benchmark models, since they are the most popular ones. Besides, multivariate probabilistic ensemble (MuPEn) [31] is adopted as a naive benchmark. It is a generalized model of the complete-history persistence, which conducts random sampling without replacement from historical scenarios for each test data.

D. Implementation Details

1) Univariate Cases: The base distributions of NN-G, NN-L, and the proposed model are set as Gaussian distribution. We carry out preliminary tests to determine the hyperparameters of the proposed model, which are presented in the appendix. The NN that determines shape parameters of the Gaussian distributions contains 2 hidden fully connected layers (each has 512 units). For fairness, we use the same amount of transforms (concretely, 5 transforms here) for NN-G, NN-L, and the proposed model. All the transforms are implemented by NNs with 2 hidden fully connected layers, each of which contains 256 units. Such transforms in the proposed model are specified as neural spline transforms, whereas they are are designed as affine transforms in the NN-G and NN-L. particularly, for NN-L, we use a sigmoid transform behind the 5 affine transforms.

2) Multivariate Cases: For multivariate cases, NN-G, NN-L and the proposed model use the same NN architecture used for univariate cases. The only difference is that we adopt the auto-regressive structure here to model the joint probability density. It is implemented based on masked auto-encoder [32] that forces each variable to only rely on the preceding variables in a given order via masks. Besides, we permute variable orders after each transform, as PDF is permutation invariant.

NN-G, NN-L, and the proposed model are established via Pytorch and trained by Adam optimizer [33]. The learning rate is determined through a grid search and ultimately set as 1e-4. It decays 1/3 per 300 epochs. The QRGBM is implemented based on lightGBM[4], the hyperparameters of which are set according to the winner of GEFCom 2014 [9]. KDE is implemented by using scikit-learn[5].

V. Results and Discussion

A. Case 1

1) CRPS: Values of CRPS are presented in Table III. It is seen that all the benchmark models and the proposed model outperform climatology model. Amongst the benchmark models, KDE has slightly lower performance than others, which suggests that it is overly simplified to approximate the

[4]https://lightgbm.readthedocs.io/en/latest/
[5]https://scikit-learn.org/stable/

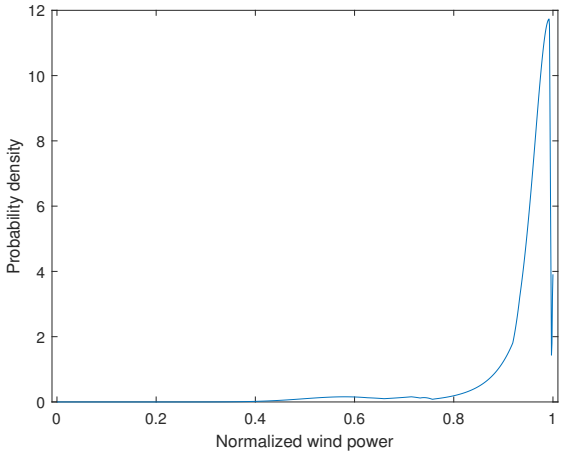

Fig. 4: Illustration of probability density of 100-th sample in test set

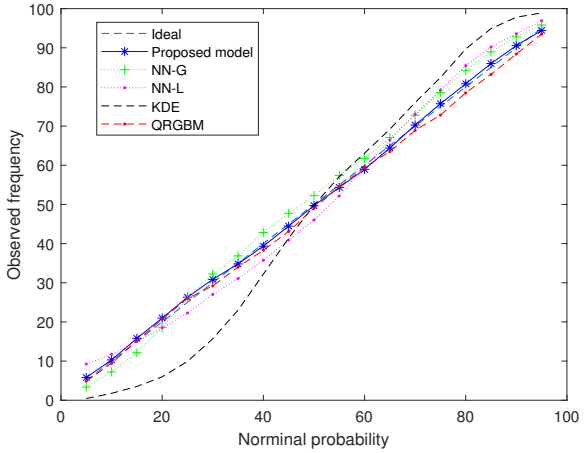

Fig. 5: Reliability diagram of forecasts at wind farm 1

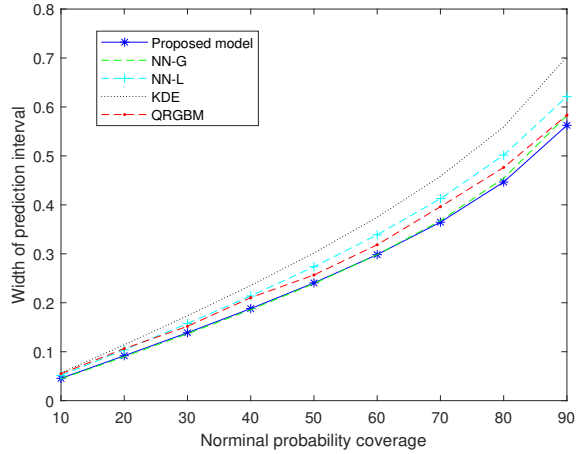

Fig. 6: Width of PI at wind farm 1

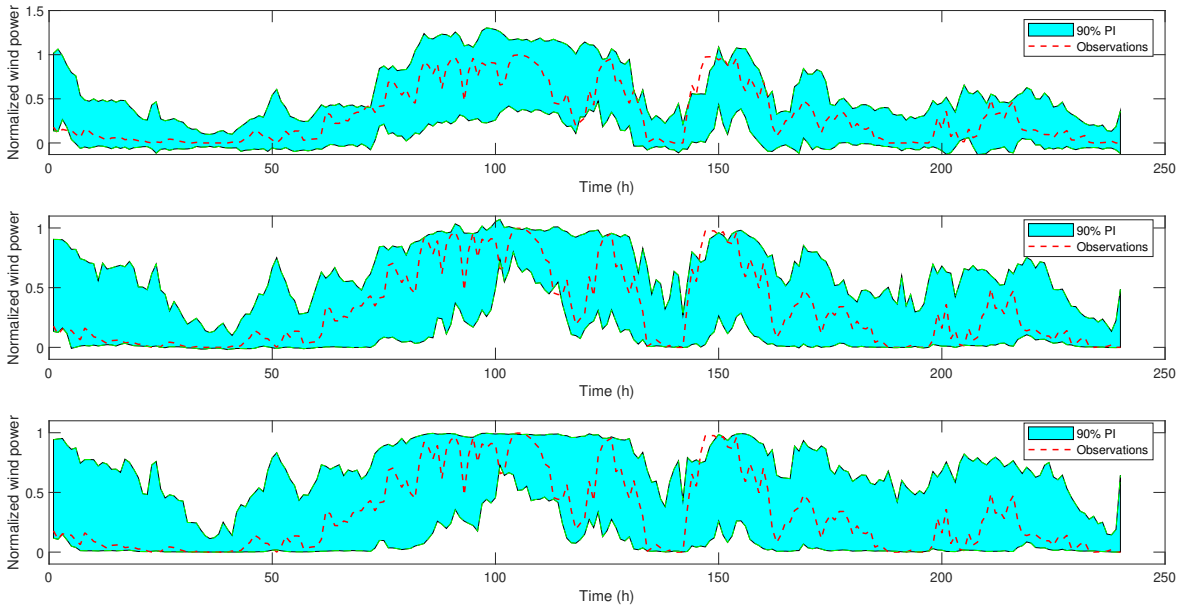

Fig. 7: 90% PI of the proposed model of 10 days at wind farm 1, top: NN-G, middle: proposed, bottom: NN-L.

TABLE IV: CRPS for both French wind farm and NREL dataset (percentage of nominal capacity)

	Climatology	NN-G	NN-L	QRGBM	KDE	Proposed model
French wind farm	13.40	1.85	1.82	1.92	3.17	1.83
NREL	21.96	0.25	0.25	0.34	2.58	0.28

conditional PDF by the density of neighborhood population. Concretely, the distribution of samples is not homogeneous, which means more samples could be taken to more accurately estimate the conditional PDF if the neighborhood distribution is dense. However, once the criterion to select neighborhood samples is fixed, e.g. value of k in k-nearest neighbors here, it cannot adaptively adjust the population, on which the conditional PDF estimation is based. On the contrary, NN-G, NN-L, QRGBM, and the proposed model can adaptively estimate the conditional PDF/quantile by excavating the similarity of input features via parameterization or entropy measure. It also suggests that the performance of KDE can be further improved by carefully designing such population selection criterion and making use of the similarity of neighborhood samples. The QRGBM outperforms the NN-G and NN-L in 3/5 cases as it is distribution-free. However, the other two cases suggest that the independent fitting in QRGBM may accumulate errors. Obviously, the proposed model exceeds benchmarks in all cases.

The comparison between NN-L and NN-G shows that logit-normal transform can degenerate performance at times. It reveals that the logit-normal distribution assumption may not match the real distribution sometimes, although the variable is forced to fall into the physical constraints. We present the predictive probability density of the proposed model at a selected timestamps in Fig. 4. As illustrated, the PDF formats

TABLE V: ES for Case 3 and Case 4 (percentage of nominal capacity)

	MuPEn	NN-G	NN-L	Proposed model
Case 3	33.73	9.32	9.20	9.20
Case 4	51.95	2.68	2.64	2.44

TABLE VI: VS in Case 3 and Case 4

	MuPEn	NN-G	NN-L	Proposed model
Case 3	0.3842	0.2711	0.2377	0.2424
Case 4	0.6303	0.0634	0.0524	0.0446

of the proposed model are more flexible than specific families of distributions because the proposed model does not rely on any assumptions about the distribution. In addition, the proposed model has 1.7 million trainable parameters, which are comparable to those of NN-G and NN-L, i.e. 1.6 million trainable parameters. This means that the proposed model can flexibly model different wind power distribution characteristics on condition of predicted wind speeds, with increased but affordable complexity.

2) Reliability and Sharpness: The reliability diagram and PI width for wind farm 1 are illustrated in Fig. 5 and Fig. 6. It turns out that QRGBM and the proposed model achieve the

TABLE VII: Training time of all models in Case 1 (seconds)

Models	NN-G	NN-L	QRGBM	KDE	Proposed
Time	56	56	44	14	78

best performance in reliability, which are close to the ideal case. Strictly speaking, it is unfair to compare a bunch of independently trained QR models with a single model that derives the whole distribution, as the computational cost of QR for a single quantile is much larger. Nevertheless, the proposed model still achieves comparable reliability, which conforms its performance. By contrast, the reliability diagrams of NN-G, NN-L, and KDE deviate from the ideal to a certain degree. The deviation of NN-G and NN-L cannot be totally mitigated, since the families of distribution they define mismatch the real underlying distribution. Results suggest that the superiority of the proposed model goes beyond the distribution-free property compared to the QR and KDE-based methods, by offering an efficient and continuous conditional modeling approach.

Fig. 6 demonstrates that the proposed model provides the shortest PI in all nominal levels. However, the performance of NN-G in width of PI is comparable to that of the proposed model, whereas the PI width of NN-L is much wider. For illustration, we present 90% PI of the NN-G, proposed model, and NN-L in 10 days in the top, middle, and bottom subplots of Fig. 7. As shown, the PI of NN-G violate the bounds of wind power to a large extent, revealing probability leakage issue, while PIs of the proposed model and NN-L are more realistic. Besides, it is demonstrated that PIs of NN-L are sometimes unnecessarily wide. For example, between 200-h and 250-h, the upper bound of NN-L is larger than that of NN-G and the proposed model. Indeed, both the NN-L and the proposed model can be considered as models derived from the NN-G by applying transforms. Indeed, the logit-normal transform in the NN-L applies to the whole domain indifferently, whereas the spline transform of the proposed model is piece-wise. This explains the sacrifice of NN-L in PI width, which is a side-effect when forcing the variable into the boundary.

B. Case 2

We present the CRPS of Case 2 in Table IV. As with Case 1, all models are superior to climatology. The performance of NN-G, NN-L, QRGBM and the proposed model are demonstrated to be comparable. The gap of performance between the KDE and others is enlarged compared to Case 1, because of higher dimension of input features which raises issues for K-nearest neighbors. Comparing results of KDE, QRGBM, and the proposed model with results of NN-G and NN-L, we can infer that the assumption of Gaussian distribution and logit-normal distribution is fairly adequate in very-short-term PWPF. This may be due to the fact that the structure of temporal interdependence over a short period of time is simpler than the interdependence spanning several hours.

C. Case 3 and Case 4

The ES and VS are presented in Table V and Table VI. All of NN-G, NN-L, and the proposed model outperform MuPEn, since the MuPEn draws samples from the empirical unconditional distribution whereas other models draws samples from the estimated conditional distribution. Except for the MuPEn, the ES and VS in Case 3 are larger than those in Case 4, which indicates larger uncertainty in Case 3. This is is caused by the fact that wind power generation uncertainty increases as look-ahead time increase. In both cases, NN-L and the proposed model exceed NN-G, which suggests the limited capability of the Gaussian assumption in complex and high dimensional cases. Besides, the performance of NN-L and the proposed model are comparable in Case 3, but they differ in Case 4, which suggests that spatial interdependence is more complex.

D. Distribution-free vs Assumption on Specific Distribution

In the case study, QRGBM, KDE, and the proposed model are distribution-free, whereas NN-G and NN-L rely on assumptions about specific distributions. Compared to NN-G and NN-L, the proposed model has increased but affordable complexity due to its spline operation. Meanwhile, the increased complexity enables the proposed model to adapt to different wind power distribution characteristics at different predicted wind speeds. Compared to QRGBM and KDE, the proposed model is superior in efficiently modeling whole conditional PDFs. In addition, case studies show that distribution-free methods are not overwhelmingly superior to models with distribution assumptions. Concretely, NN-G and NN-L rival QRGBM and KDE in several cases. And in Case 2, the performance of NN-L is comparable to that of the proposed model, which means these assumptions are adequate in very-short-term PWPF. But when it comes to scenarios with more uncertainty and more complex interdependence, the proposed approach always achieves a satisfactory performance with an acceptable computational cost.

E. Training Time

Training time of all models in Case 1 is presented in Table VII, we report the training time of 1000 epochs of NN-based models and 199 independent quantiles of QRGBM. It shows that the training time of the proposed model is comparable to that of commonly used NN-G, which is affordable. In general, the training time of the proposed model is governed by the number of transforms and the number of hidden units. With more transforms and hidden units, the training time will increase.

VI. Conclusions

The approach for probabilistic wind power forecasting described in this paper, based on conditional normalizing flow, offers a number of advantages with respect to the existing. It directly estimates the conditional probability density and does not require any assumption on the distributions involved. In addition, it is applicable to both univariate PWPF and multivariate PWPF, with high efficiency in terms of both modeling and computing. Our case-study applications based on open datasets confirmed the interest of the approach and its wide applicability for wind power applications.

TABLE VIII: CRPS under different steps of transforms (percentage of nominal capacity)

Number of Transforms	1	2	3	4	5
CRPS	9.93	9.51	9.23	9.25	9.22

TABLE IX: CRPS under different sizes of hidden units (percentage of nominal capacity)

Number of Units	64	256	512
CRPS	9.22	9.08	9.37

TABLE X: CRPS under different number of knots (percentage of nominal capacity)

Knots	5	10	20	50
CRPS	9.25	9.08	9.19	9.20

APPENDIX

A. Selection on Hyperparameters

To empirically determine the hyperparameters, we conduct a preliminary test to validate the influence of number of transforms, number of units, and number of knots by studying variants of Case 1. Specifically, we take wind farm 1 as an example, and present results of several case settings.

1) Number of Transforms: In this case, we set the number of hidden units in transform as 64, the number of knots as 10, and vary the number of transforms from 1 to 5. The corresponding results are shown in Table VIII. It can be seen that the CRPS is relatively larger when we use only few transforms. Consequently, the model is small, which results in limited capability of fitting ultimate transform and shape parameter function of base distribution. After reaching at 3 transforms, the gain of increasing transforms is relatively low, which suggest the capability is enough. Besides, increasing transforms means increasing layers of deep neural network, whose training procedure might become difficult when the model is considerably deep.

2) Number of Hidden Units: Here we fix the number of transforms as 5, the number of knots as 10, and adjust the number of hidden units as 64, 256, and 512. Results are presented in Table IX. It shows that the fitting capability of NN in each transform is influenced by the number of hidden units. The capability is limited when the number of hidden units is few. But it might overfit the data if the number of hidden units is considerable.

3) Number of Knots: In this case, we fix the number of layers as 5 and the number of hidden units as 256, and look into the influence of knots by varying the number. We set it as 5, 10, 20, and 50 respectively, whose results are shown in Table X. As we increase the number of knots, the CRPS first decreases and then increases.

ACKNOWLEDGMENT

This work was performed during a research stay at the Technical University of Denmark. The authors would like to appreciate China Scholarship Council (NO. 202006230261) and Shanghai Sailing Program (19YF1423700). The research leading to this work is being carried out as a part of the Smart4RES project (European Union's Horizon 2020, No. 864337). The sole responsibility of this publication lies with the authors. The European Union is not responsible for any use that may be made of the information contained therein.

REFERENCES

[1] C. Sweeney, R. J. Bessa, J. Browell, and P. Pinson, "The future of forecasting for renewable energy," *Wiley Interdisciplinary Reviews: Energy and Environment*, vol. 9, no. 2, p. e365, 2020.

[2] J. M. Morales, A. J. Conejo, H. Madsen, P. Pinson, and M. Zugno, *Integrating renewables in electricity markets: operational problems*. Springer Science & Business Media, 2013, vol. 205.

[3] M. Lange, "On the uncertainty of wind power predictions—analysis of the forecast accuracy and statistical distribution of errors," *J. Sol. Energy Eng.*, vol. 127, no. 2, pp. 177–184, 2005.

[4] P. Pinson and G. Kariniotakis, "Conditional prediction intervals of wind power generation," *IEEE Transactions on Power Systems*, vol. 25, no. 4, pp. 1845–1856, 2010.

[5] J. B. Bremnes, "A comparison of a few statistical models for making quantile wind power forecasts," *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, vol. 9, no. 1-2, pp. 3–11, 2006.

[6] P. Pinson, H. Madsen, H. A. Nielsen, G. Papaefthymiou, and B. Klöckl, "From probabilistic forecasts to statistical scenarios of short-term wind power production," *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, vol. 12, no. 1, pp. 51–62, 2009.

[7] J. Tastu, P. Pinson, and H. Madsen, "Space-time trajectories of wind power generation: Parametrized precision matrices under a gaussian copula approach," in *Modeling and stochastic learning for forecasting in high dimensions*. Springer, 2015, pp. 267–296.

[8] C. Wan, J. Lin, J. Wang, Y. Song, and Z. Y. Dong, "Direct quantile regression for nonparametric probabilistic forecasting of wind power generation," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2767–2778, 2016.

[9] M. Landry, T. P. Erlinger, D. Patschke, and C. Varrichio, "Probabilistic gradient boosting machines for gefcom2014 wind forecasting," *International Journal of Forecasting*, vol. 32, no. 3, pp. 1061 – 1066, 2016.

[10] C. Zhao, C. Wan, and Y. Song, "An adaptive bilevel programming model for nonparametric prediction intervals of wind power generation," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 424–439, 2019.

[11] Y. Zhang, J. Wang, and X. Wang, "Review on probabilistic forecasting of wind power generation," *Renewable and Sustainable Energy Reviews*, vol. 32, pp. 255–270, 2014.

[12] Y. Zhang and J. Wang, "K-nearest neighbors and a kernel density estimator for gefcom2014 probabilistic wind power forecasting," *International Journal of forecasting*, vol. 32, no. 3, pp. 1074–1080, 2016.

[13] M. Afrasiabi, M. Mohammadi, M. Rastegar, and S. Afrasiabi, "Advanced deep learning approach for probabilistic wind speed forecasting," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 720–727, 2020.

[14] O. Makansi, E. Ilg, O. Cicek, and T. Brox, "Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7144–7153.

[15] Y. Chen, Y. Wang, D. Kirschen, and B. Zhang, "Model-free renewable scenario generation using generative adversarial networks," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 3265–3275, 2018.

[16] C. Chu, K. Minami, and K. Fukumizu, "Smoothness and stability in gans," in *International Conference on Learning Representations*, 2019.

[17] J. Dumas, A. Wehenkel, D. Lanaspeze, B. Cornélusse, and A. Sutera, "A deep generative model for probabilistic energy forecasting in power systems: normalizing flows," *Applied Energy*, 2021, in press, available online.

[18] P. Pinson, "Very-short-term probabilistic forecasting of wind power with generalized logit–normal distributions," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 61, no. 4, pp. 555–576, 2012.

[19] E. M. Stein and R. Shakarchi, *Princeton lectures in analysis*. Princeton University Press, 2003.

[20] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *International conference on machine learning*. PMLR, 2015, pp. 1530–1538.

[21] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference," *Journal of Machine Learning Research*, vol. 22, no. 57, pp. 1–64, 2021.
[22] G. Papamakarios, T. Pavlakou, and I. Murray, "Masked autoregressive flow for density estimation," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 2335–2344.
[23] C. Winkler, D. Worrall, E. Hoogeboom, and M. Welling, "Learning likelihoods with conditional normalizing flows," *arXiv preprint arXiv:1912.00042*, 2019.
[24] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios, "Neural spline flows," *Advances in Neural Information Processing Systems*, vol. 32, pp. 7511–7522, 2019.
[25] A. Kupers, "Lecture notes on diffeomorphism groups of manifolds," Harvard University, February 2019.
[26] A. Khosravi, S. Nahavandi, S. Member, D. Creighton, A. F. Atiya, and S. Member, "Comprehensive Review of Neural Network-Based Prediction Intervals and New Advances," *IEEE Transactions on Neural Networks*, vol. 22, no. 9, pp. 1341–1356, 2011.
[27] P. Kou, F. Gao, and X. Guan, "Sparse online warped gaussian process for wind power probabilistic forecasting," *Applied energy*, vol. 108, pp. 410–428, 2013.
[28] J. W. Messner, P. Pinson, J. Browell, M. B. Bjerregård, and I. Schicker, "Evaluation of wind power forecasts—an up-to-date view," *Wind Energy*, vol. 23, no. 6, pp. 1461–1481, 2020.
[29] P. Pinson and R. Girard, "Evaluating the quality of scenarios of short-term wind power generation," *Applied Energy*, vol. 96, pp. 12–20, 2012.
[30] M. Scheuerer and T. M. Hamill, "Variogram-based proper scoring rules for probabilistic forecasts of multivariate quantities," *Monthly Weather Review*, vol. 143, no. 4, pp. 1321–1334, 2015.
[31] D. van der Meer, "A benchmark for multivariate probabilistic solar irradiance forecasts," *Solar Energy*, vol. 225, pp. 286–296, 2021.
[32] M. Germain, K. Gregor, I. Murray, and H. Larochelle, "Made: Masked autoencoder for distribution estimation," in *International Conference on Machine Learning*. PMLR, 2015, pp. 881–889.
[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.