A Composite Quantile Fourier Neural Network for Multi-Horizon Probabilistic Forecasting

Kostas Hatalis and Shalinee Kishore

Abstract

A novel quantile Fourier neural network is presented for nonparametric probabilistic forecasting. Prediction are provided in the form of composite quantiles using time as the only input to the model. This effectively is a form of extrapolation based quantile regression applied for forecasting. Empirical results showcase that for time series data that have clear seasonality and trend, the model provides high quality probabilistic predictions. This work introduces a new class of forecasting of using only time as the input versus using past data such as an autoregressive model. Extrapolation based regression has not been studied before for probabilistic forecasting.

Index Terms

Nonparametric probabilistic forecasting, Fourier extrapolation, artificial neural networks, composite quantile regression.

I. Introduction

Time series based deterministic or point forecasting is a well studied field that has numerous applications. There are several approaches to forecasting with different classes of methods. Approaches include having a sliding window of past data to predict future data, recurrent neural network models, and extrapolation based regression such as signal approximation. In all the approaches, methods can be divided in two classes as linear or nonlinear. In the first class, models include linear regression models, autoregressivemoving-average (ARMA) models, exponential smoothing, and so forth. The second class of methods are nonlinear models which are predominantly machine learning based such as support vector regression, nonlinear autoregression neural networks, recurrent neural networks such as LSTMs, etc. Deterministic forecasts which provide a single expected output for a given look-ahead time have been successfully applied for multiple domains such as renewable energy prediction for solar, wind, and wave power. Other applications include the smart grid, economics, finance, and manufacturing. A thorough overview of time series and machine learning based deterministic forecasting can be found in [1], [2].

Despite the popularity of deterministic forecasting it does have a big disadvantage in that it can result in certain errors which can be significant. Additionally, deterministic forecasting also lacks information on associated uncertainty. To address these problems, a probabilistic forecast (PF) can be applied to produce fully probabilistic predictions that derive quantitative information on the uncertainty. A PF takes the form of a predictive probability distribution over future time horizons. PF aims to maximize the sharpness of the predictive density, subject to reliability, on the basis of available data. Proper scoring rules are used to assess reliability and sharpness simultaneously.

We highlight specific applications of PF to the fields of renewable energies and power systems. A probabilistic forecast is vital, for different operations to renewable energy farms. This includes managing the optimal level of generation reserves [3], optimizing production [4], and bidding strategies for electricity markets [5]. Applications to the power grid include load analysis [6], smart meters [7], scheduling [8], system planning [9], unit commitment [10], and energy trading [11]. A thorough overview of probabilistic wind and solar power forecasting is provided in [12] and [13].

There are several important classes in the type of PF models which include if they are parametric or nonparametric, direct or indirect, and the type of inputs they use for prediction. In PF, we are first trying to predict one of two classes of density functions, either parametric or nonparametric. When the future density function is assumed to take a certain distribution, such as the Normal distribution, then this is known as parametric probabilistic forecasting. For processes where no assumption is made about the shape of the distribution, a nonparametric probabilistic forecast can be made. Nonparametric predictions can be made in the form of quantiles, prediction intervals, or full density functions. For example, nonlinear and non-stationary data such as wind speeds, they may not correspond to fixed distributions. So when in need of forecasting such data it is more beneficial to apply a nonparametric probabilistic forecast to estimate the distribution rather then assume it's shape.

The second classification of PF models is whether they are combined with point forecasts or not. For instance in [14] a deterministic and probabilistic forecast for wind power is studied. This is known as an indirect PF method which first implements a point forecast, such as with support vector regression, and then prediction intervals for point forecast values are obtained with a PF method such as quantile regression. On the other hand when a probabilistic forecasting method estimates future quantiles or prediction intervals without using as input point forecasts, this is known as direct forecasting. The last distinction to be made with PF models, particularly for renewable energies, is if past lagged data are used as inputs to the forecast or if future exogenous variables are used. Typically, for renewable power, if numerical weather predictions (NWP) are provided for each forecasting horizon that we are interested in, then those NWPs are used as inputs to provide a PF in that prediction horizon. When NWPs are not given then we can use lagged past time values of renewable power.

We introduce a new approach to developing a nonparametric direct PF where the input to the model is neither NWPs, in the case of renewable forecasting, nor past data but rather treats the series as a signal. This approach is motivated by Fourier extrapolation which is the process by which a Fourier transform is applied to a data set to decompose it into a sum of sinusoidal components thus interpreting it as a signal. In time series analysis this is related to Harmonic regression. In accounting for periodic and non-periodic aspects of a signal such as trend, Fourier neural networks (FNN) have been proposed. FNNs are feedforward neural networks with sinusoidal activation functions that model the Fourier transform. Most recently, a new FNN called neural decomposition (ND) was introduced in [15] that is able to decompose a signal into a sum of its constituent parts, model trend, and reconstruct a signal beyond the training samples. ND is able to provide a prediction by having time as its only input similarly to an inverse Fourier transform. Our approach follows the ND model.

Several works have explored Fourier extrapolation based deterministic forecasting with sinusoidal neural networks, but none have explored it for probabilistic forecasting. We are the first to introduce a FNN for forecasting composite quantiles that we dub the composite quantile Fourier neural network. The main contributions of our approach can be summarized as follows:

1) We show how to estimate composite quantiles using a Fourier neural network that is able to model seasonality and trend of a time series.
2) We demonstrate a simple weight initialization process that fixes parameters to none random values, and train the model with gradient descent backpropagation.

3) We design experiments to validate our approach for direct probabilistic forecasting and provide insight how this method is able to generalize modeling uncertainty on real-world datasets.

The contents of the paper are: in Section II we provide the mathematical background on probabilistic forecasting, quantile regression, and evaluation methods. In Section III we go over our model, its architecture, training, and weighting initialization scheme. Results and discussion of our case study are presented in Section IV.

II. Background on Density Forecasting

This section highlights the underlying mathematics in probabilistic forecasting, overviews linear quantile regression, and summarizes the main evaluation metric for density forecasts. Given a random variable Y_t such as solar power at time t, its probability density function is defined as f_t and its the cumulative distribution function as F_t. If F_t is strictly increasing, the quantile $q_t^{(\tau)}$ at time t of the random variable Y_t with nominal proportion τ is uniquely defined on the value x such that $P(Y_t < x) = \tau$ or equivalently as the inverse of the distribution function $q_t^{(\tau)} = F_t^{-1}(\tau)$. A quantile forecast $\hat{q}_{t+z}^{(\tau)}$ is an estimate of the true quantile $q_{t+z}^{(\tau)}$ for the lead time $t+z$, given a predictor values. Prediction intervals are another type of probabilistic forecast and give a range of possible values within which an observed value is expected to lie with a certain probability $\beta \in [0,1]$. A prediction interval $\hat{I}_{t+z}^{(\beta)}$ produced at time t for future horizon $t+z$ is defined by its lower and upper bounds, which are the quantile forecasts $\hat{I}_{t+z}^{(\beta)} = \left[\hat{q}_{t+z}^{(\tau_l)}, \hat{q}_{t+z}^{(\tau_u)}\right] = \left[l_t^{(\beta)}, u_t^{(\beta)}\right]$ whose nominal proportions τ_l and τ_u are such that $\tau_u - \tau_l = 1 - \beta$.

In probabilistic forecasting, we are trying to predict one of two classes of density functions, either parametric or nonparametric. When the future density function is assumed to take a certain distribution, such as the Normal distribution, then this is called parametric probabilistic forecasting. For a nonlinear and bounded process such as wind generation, probability distributions of future wind power, for instance, may be skewed and heavy-tailed distributed [16]. Else if no assumption is made about the shape of the distribution, a nonparametric probabilistic forecast $\hat{f}_{t+z}$ [17] can be made of the density function by gathering a set of M quantiles forecasts such that $\hat{f}_{t+z} = \left\{\hat{q}_{t+z}^{(\tau_m)}, m = 1, ..., M | 0 \le \tau_1 < ... < \tau_M \le 1\right\}$ with chosen nominal proportions spread on the unit interval.

Quantile regression is a popular approach for nonparametric probabilistic forecasting. Koenker and Bassett [18] introduce it for estimating conditional quantiles and is closely related to models for the conditional median [19]. Minimizing the mean absolute function leads to an estimate of the conditional median of a prediction. By applying asymmetric weights to errors through a tilted transformation of the absolute value function, we can compute the conditional quantiles of a predictive distribution. The selected transformation function is the pinball loss function as defined by

$$\rho_\tau(u) = \begin{cases} \tau u & \text{if } u \ge 0 \\ (\tau - 1)u & \text{if } u < 0 \end{cases}, \tag{1}$$

where $0 < \tau < 1$ is the tilting parameter. To better understand the pinball loss, we look at an example for estimating a single quantile. If an estimate falls above a reported quantile, such as the 0.05-quantile, the loss is its distance from the estimate multiplied by its probability of 0.05. Otherwise, the loss is its distance from the realization multiplied by one minus its probability (0.95 in the case of the 0.05-quantile). The pinball loss function penalizes low-probability quantiles more for overestimation than for underestimation and vice versa in the case of high-probability quantiles. Given a vector of predictors X_t where $t = 1, ..., N$, a vector of weights W and intercept b coefficient in a linear regression fashion, the conditional τ quantile is given by $\hat{q}_t^{(\tau_i)} = W^\top X_t + b$. To determine estimates for the weights and intercept for composite quantile regression we solve the following minimization problem

$$\min_{W,b} \frac{1}{NM} \sum_{i=1}^{M} \sum_{t=1}^{N} \rho_\tau(y_t - \hat{q}_t^{(\tau_i)}), \tag{2}$$

where y_t is the observed value of the predictand and M is the number of quantiles we're estimating. The formulation above in Eq. (2) can be minimized by a linear program.

A. Evaluation Methods

In probabilistic forecasting it is essential to evaluate the quantile estimates and if desired also evaluate derived predictive intervals. To evaluate quantile estimates, one can use the pinball function directly as an assessment called the quantile score (QS). We choose QS as our main evaluation measure for the following reasons. When averaged across many quantiles it can evaluate full predictive densities; it is found to be a proper scoring rule [20]; it is related to the continuous rank probability score; and it is also the main evaluation criteria in the 2014 Global Energy Forecasting Competition (GEFCOM 2014), the source of our testing data. QS calculated overall N test observations and M quantiles is defined as

$$QS = \sum_{t=1}^{N} \sum_{m=1}^{M} \rho_{\tau_m}(y_t - \hat{q}_t^{(\tau_m)})$$

where y_t is an observation used to forecast evaluation. To evaluate full predictive densities, this score is averaged across all target quantiles for all look ahead time steps using equal weights. A lower QS indicates a better forecast.

III. Model Description

Fourier analysis examines the approximation of functions through their decomposition as a sum or product of trigonometric functions, while Fourier synthesis focuses on the reconstruction of a signal from it's decomposed oscillatory components. These well studied processes have a large utility in time series analysis. By decomposing a time series into it's frequencies one could then interpolate missing time values by reconstructing the original signal. Further applications include modeling seasonality and even prediction of a time series through extrapolation of an approximated signal. In the application of Fourier analysis for time series analysis, an important method is the discrete Fourier transform (DFT), which converts a series into it's frequency domain representation, and the inverse discrete Fourier transform (iDFT) which maps the frequency representation back to the time domain. The transforms can be expressed as either a summation of complex exponentials or of sines and cosines by Eulers formula. In this section we explore existing works on Fourier networks that directly use iDFT in their operation or mimic it, then we describe our FNN approach for quantile forecasting.

A. Fourier Neural Networks

Neural networks with sine as an activation function are difficult to train in theory and when initialized randomly yield poor results [21]. Thus, few works have attempted to explore modeling Fourier transforms with sinusoidal neural networks. We highlight most of the works here. One of the first FNNs was introduced by Adrian Silvescu [22], [23] who developed Fourier-like neurons for learning boolean functions. The FNN model used the units of the network to approximate a DFT in its output. Similar in spirit to a FNN was a Fourier transform neural network introduced in [24] that uses the Fourier transform of the data as input to an artificial neural network. FNNs have since been used for stock prediction [25], aircraft engine fault diagnostics [26], harmonic analysis [27], and extensions include a single input multiple outputs based FNNs that can turn nonlinear optimization problems into linear ones [25], FNNs for output feedback learning control schemes [28], and deep FNNs for lane departure prediction [29].

There are two recent works that study FNNs for time series prediction that use the Fourier transform of the data as weights. The first is a FNN presented by Gashler and Ashmore in [30]. Their technique uses the fast Fourier transform (FFT) to approximate the DFT and then uses the obtained values to initialize the weights of the neural network. Their model uses a combination of sinusoid, linear and softplus activation units for modeling periodic and non-periodic components of a time series. However, their trained models were slightly out of phase with their validation data. The second study on FNNs for time series prediction was presented by Godfrey and Gashler [15] who proposed a similar model to [30] called neural decomposition (ND), except that they do not use the Fourier transform to directly initialize any weights.

The ND network is inspired by the inverse discreet Fourier transform (iDFT) where given time t as input it attempts to model the signal $x(t)$. However, there are some distinctions. First ND allows sinusoid frequencies to be trained and second ND can also model non-periodic components in a signal such as trend. With the ability to train the frequencies ND learns the true period of a signal whereas iDFT assumes that the underlying function always has a period equal to the size of the samples it represents. ND is a feedforward neural network with a single hidden layer with N nodes and has one input and one output node. Hidden nodes are composed of sinusoid units for capturing the periodic component in an underlying signal and other activation functions, such as linear or sigmoid units, for capturing the non-period component. Parameters of ND are initialized in such a way so as to mimic the iDFT. ND is then trained with stochastic gradient descent with backpropagation and uses L1 regularization to promote sparsity by driving nonessential weights to zero. ND was applied to time series deterministic forecasting and showed very promising results across different data sets, often beating state-of-the-art methods such as LSTM, SVR, and SARIMA.

B. Quantile Fourier Neural Networks

Motivated by the ND model we propose a new forecasting method which we call the quantile Fourier neural network (QFNN). Our model allows sinusoid frequencies to be trained in order to capture cycles in a time series and also has a non-periodic function to model components such as trend. Unlike ND and other FNNs our QFNN model is trained to estimate quantiles of an underlying time series. The use of sinusoid activation functions allows the model to fit periodic data, and coupled with a trend function QFNN is able to probabilistically forecast time series that are made up of both periodic and non-periodic

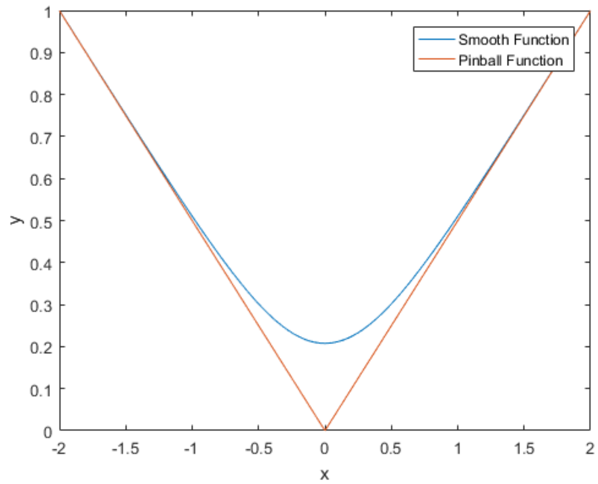

Fig. 1. Pinball ball function versus the smooth pinball neural network with smoothing parameter $\alpha = 0.2$.

components. The model is defined as follows. Let each a_{jk} represent an amplitude, each w_k represent a frequency, and each ϕ_k represent a phase shift. Let $f(t)$ be an augmentation trend function that represents the non-periodic components of the signal. QFNN then can be defined by

$$q_t^\tau = f(t) + \sum_{k=1}^{H} a_{\tau,k} \cdot \cos\left(\omega_k t + \phi_k\right) \tag{3}$$

where given time as the input, it attempts to predict the τ-level quantile.

QFNNs hidden layer is composed of N units with a sinusoid activation function and an arbitrary number of units with other activation functions to calculate $f(t)$. The output layer is composed of M number of linear units that represent quantiles. The parameters a_{jk}, being the weights between the hidden and output layers allows us to model different amplitudes for composite quantiles while simultaneously learning the frequency and phases for all quantiles in the hidden layer.

To estimate quantiles we need to solve the minimization problem described in Eq. 2. However, the pinball function ρ in Eq. 2. is not differentiable at the origin, $x = 0$. The non-differentiability of ρ makes it difficult to apply gradient-based optimization methods in fitting the quantile regression model. Gradient-based methods are preferable for training neural networks since they are time efficient, easy to implement and yield a local optimum. Therefore, we need a smooth approximation of the pinball function that allows for the direct application of gradient-based optimization. A smooth approximation [31] of the pinball function in Eq. (1) can be given by

$$S_{\tau,\alpha}(u) = \tau u + \alpha \log\left(1 + \exp\left(-\frac{u}{\alpha}\right)\right), \tag{4}$$

where $\alpha > 0$ is a smoothing parameter and $\tau \in [0, 1]$ is the quantile level we're trying to estimate. In Fig. 1 we see the pinball function with $\tau = 0.5$ as the red line and the a smooth approximation as the blue line with $\alpha = 0.2$. Zheng proves [31] that in the limit as $\alpha \to 0^+$ that $S_{\tau,\alpha}(u) = \rho_\tau(u)$. With this smooth approximation we can then define the cost minimization problem

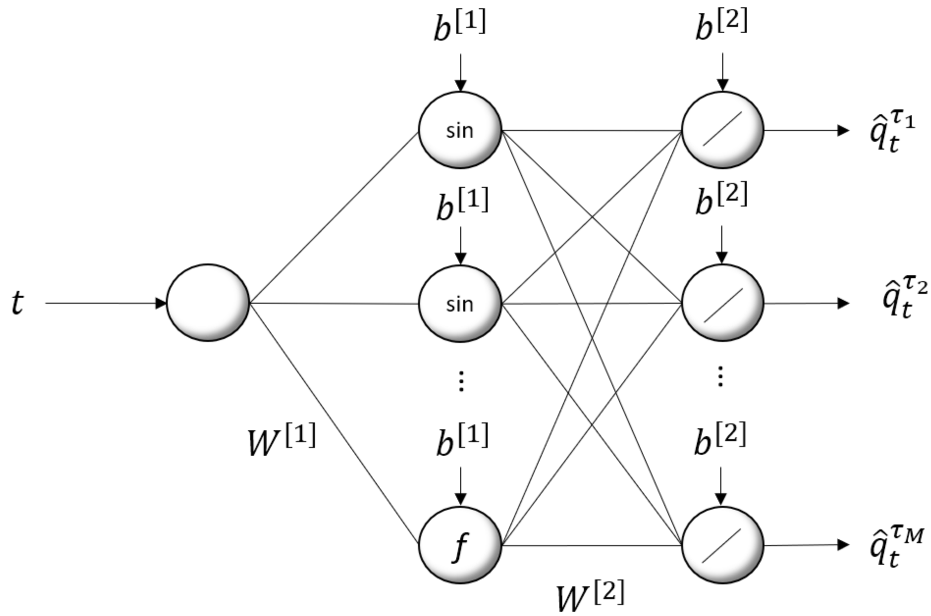

Fig. 2. Architecture of the quantile Fourier neural network.

for QFNN as

$$E = \frac{1}{NM} \sum_{t=1}^{N} \sum_{m=1}^{M} \dots \left[\tau_m (y_t - \hat{q}_t^{(\tau_m)}) + \alpha \log \left(1 + \exp \left(- \frac{y_t - \hat{q}_t^{(\tau_m)}}{\alpha} \right) \right) \right]. \tag{5}$$

where M number of τ's we are trying to estimate in the output layer. The input to hidden neurons is calculated, in vectorization notation, by

$$Z_t^{[1]} = W^{[1]} t + b^{[1]},$$

the output of the hidden layer then uses the logistic activation function

$$H_t = \cos\left(Z_t^{[1]}\right), f\left(Z_t^{[1]}\right).$$

The input to output neurons is then calculated by

$$Z_t^{[2]} = W^{[2]} H_t + b^{[2]},$$

and the output layer uses the identity activation function

$$\hat{Q}_t = Z_t^{[2]}.$$

An architectural view of the QFNN is shown in Fig. 2. Utilizing conventional neural network notation $W^{[1]}$ is a matrix of the $f(t)$ unit parameters and the frequency parameters in Eq. 3. The $b^{[1]}$ vector can be seen as the phases of the sinusoidal components, $W^{[2]}$ is a parameter matrix of the amplitudes, and we also add additional bias terms to the output nodes for each quantile with the $b^{[2]}$ vector.

C. Implementation Details

The QFNN model is trained using gradient descent with backpropagation. The training process allows the model to learn better frequencies and phase shifts so that the sinusoid units more accurately represent the seasonality of an underlying time series. Since frequencies and phase shifts are able to change, the model can learns a more reliable periodicity of the underlying series rather than assuming the period is of a predetermined fixed size. Training also tunes the weights of the trend function. Additionally, the cost function of the model uses elastic net regularization with a smooth approximation to the L1 component so that it is differentiable

$$\lambda \left(a\|w\|_1 + (1-a)\|w\|_2^2 \right). \tag{6}$$

Elastic net regularization is used to strike a balance between promoting sparsity amongst parameters and over-fitting the training data. There is a large distinction in how QFNN is initialized compared to other FNNs. Instead of randomly setting parameters or initializing them to mimic the iDFT we set all phases ϕ and amplitudes a to zero. The frequency ω is set to $2\pi\lfloor k/2 \rfloor$ where k is a specific hidden node. For the augmentation function we set $f(t)$ equal to $wx+b$. For initialization the bias b is set to 0 and w is set to the least squares estimation of Y. This is an estimation of the slope of the true trend line.

The full model implementation flowchart is shown in Fig. 3. First the data is preprocessed which includes partitioning the data into training and testing sets, followed by normalizing all training data between between 0 and 1 on the time axis. Parameters of the QFNN are then initialized as described in the previous paragraphs. Training of the model is conducted using batch gradient descent. After the max number of training epochs is reached the model is ready to be used on testing data for composite quantile estimation. Forecasts can be provided for a multi-horizon period of indefinite time steps.

IV. Results and Discussion

In validating the QFNN model for probabilistic forecasting we utilize load, wind, and solar data from the publicly available Global Energy Forecasting Competition 2014 (GEFCom2014). We also use simulated wave elevation provided from WECsim [32] for applications to future wave energy farms. Through empirical experimentation we found the following values as adequate for our model parameters: 1000 training iterations, 20 hidden sinusoidal nodes, 1 trend node, 0.1 for the learning rates, 0.01 for the smoothing rate, and 0.1 for the weight regularization terms.

Figures 4-7 showcase forecasting results for wave elevation, solar power, electric load, and wind power. As described in the implementation details section, training data has its time stamps (x-values) normalized between 0 to 1. In all figures the blue line at 1.00 seconds indicates that values to the left of the line are used for training. Values to the right of the line are test data. The red line indicates actual observed data. In each of the four cases we provide 100 composite quantile forecasts between 0.01 to 0.99, combining them together forms 50 prediction intervals. These 50 prediction intervals are shown in all the figures ranging from light to dark blue shades.

We use three standard benchmark methods [33] for density forecasting. The standard methods are the persistence model that corresponds to the normal distribution and is formed by the last 24 hours of observations, the climatology model that is based on all observations, and the uniform distribution that assumes all observations occur with equal probability. Averaged

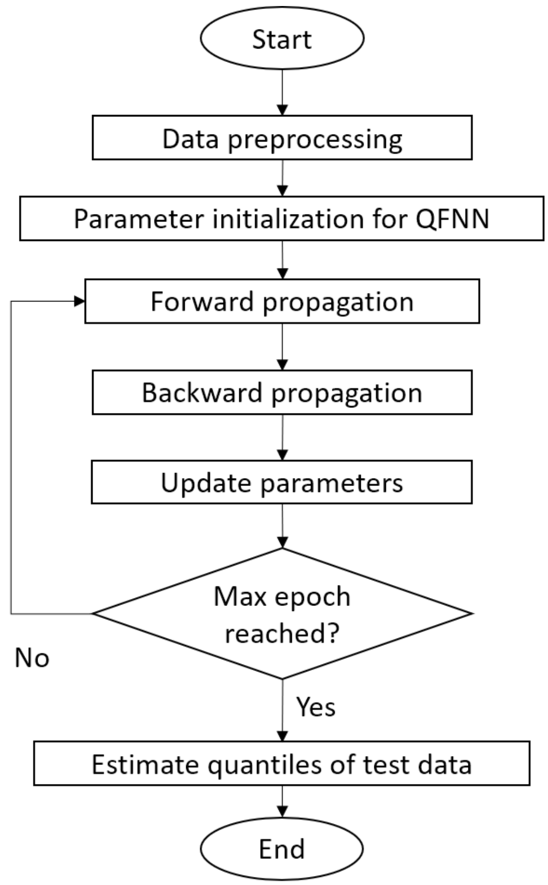

Fig. 3. Flowchart of the steps taken when conducting a probabilistic forecast with QFNN.

TABLE I
QUANTILES SCORES FROM QFNN AND BENCHMARK METHODS.

Model	Uniform	Persistence	Climatology	QFNN
Load	0.1044	0.0734	0.0701	0.0256
Wave	0.1187	0.0803	0.0791	0.0436
Wind	0.1298	0.0980	0.0953	0.0733
Solar	0.1165	0.0834	0.0816	0.0608

quantile scores for each case are shown in Table 1. Results for QFNN indicate highly reliable predictions with the lowest QS metric. Figures 4-6 show that when there is clear trend and seasonality, the QFNN model is able to provide excellent forecasts. However in more chaotic and non-stationary data such as wind power where the data is not periodic and has no clear trend, the model is not able to appropriately predict the time series.

Probabilistic predictions can provide a much better analysis of uncertainty then point forecasting. This paper introduces a novel approach for probabilistic forecasting using a quantile Fourier neural network with a smooth approximation to the pinball ball loss function in directly estimating composite quantiles. Empirical results showcase that for time series data that have clear seasonality and trend, the model provides high quality probabilistic predictions. Extrapolation based regression has not been studied before for probabilistic forecasting. This work showcases a new approach of forecasting that only uses time as the input versus using past data such as autoregressive models. Given the novelty of this forecasting approach, more research

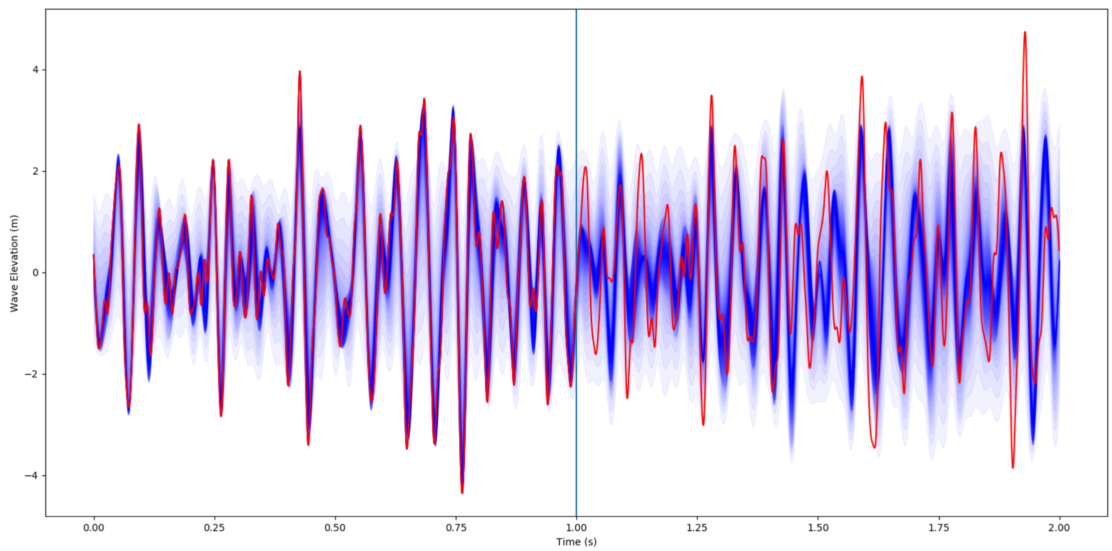

Fig. 4. Wave power probabilistic forecasting.

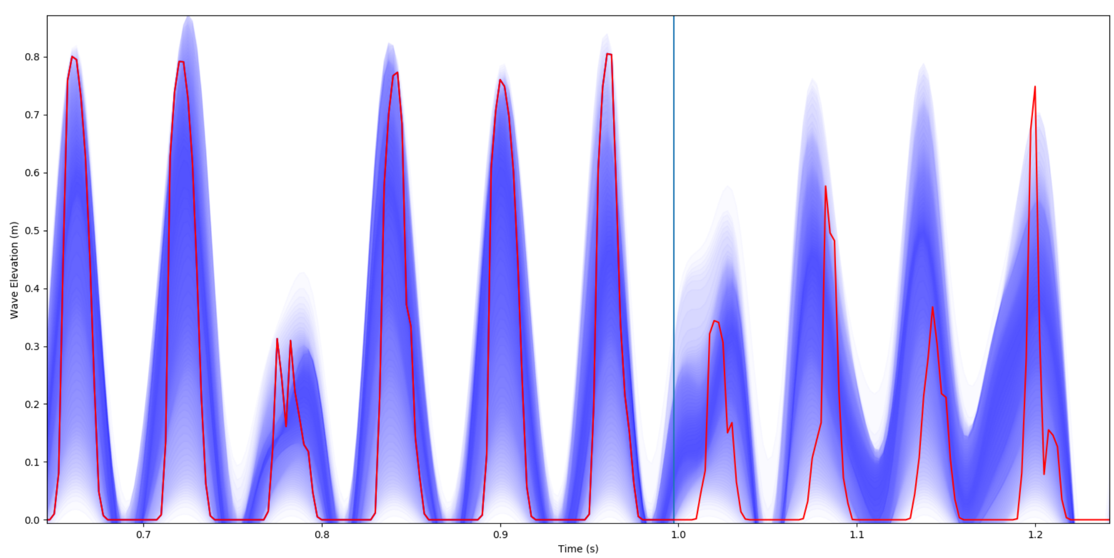

Fig. 5. Solar power probabilistic forecasting.

needs to be conducted to assess its application for more domains and under different scenarios.

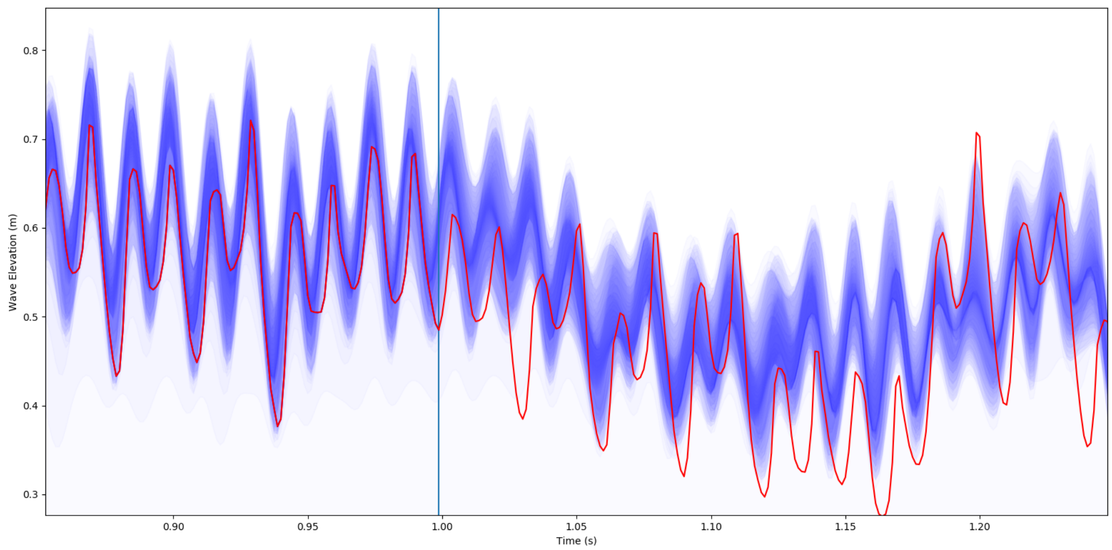

Fig. 6. Electric load probabilistic forecasting.

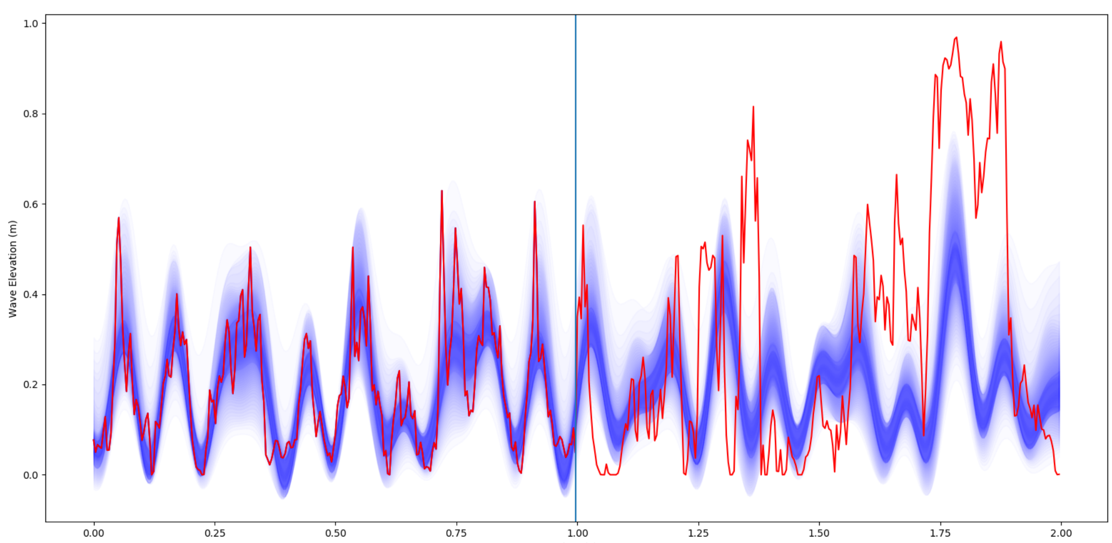

Fig. 7. Wind power probabilistic forecasting.

References

[1] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[2] M. Längkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognition Letters*, vol. 42, pp. 11–24, 2014.

[3] R. Doherty and M. O'Malley, "A new approach to quantify reserve demand in systems with significant installed wind capacity," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 587–595, 2005.

[4] E. D. Castronuovo and J. P. Lopes, "On the optimization of the daily operation of a wind-hydro power plant," *IEEE Transactions on Power Systems*, vol. 19, no. 3, pp. 1599–1606, 2004.

[5] P. Pinson, C. Chevallier, and G. N. Kariniotakis, "Trading wind generation from short-term probabilistic forecasts of wind power," *IEEE Transactions on Power Systems*, vol. 22, no. 3, pp. 1148–1156, 2007.

[6] Y. Yang, S. Li, W. Li, and M. Qu, "Power load probability density forecasting using gaussian process quantile regression," *Applied Energy*, 2017.

[7] S. B. Taieb, R. Huser, R. J. Hyndman, and M. G. Genton, "Forecasting uncertainty in electricity smart meter data by boosting additive quantile regression," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2448–2455, 2016.

[8] W. Su, J. Wang, and J. Roh, "Stochastic energy scheduling in microgrids with intermittent renewable energy resources," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1876–1883, 2014.

[9] T. Hong, J. Wilson, and J. Xie, "Long term probabilistic load forecasting and normalization with hourly information," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 456–462, 2014.

[10] A. Botterud, Z. Zhou, J. Wang, J. Sumaili, H. Keko, J. Mendes, R. J. Bessa, and V. Miranda, "Demand dispatch and probabilistic wind power forecasting in unit commitment and economic dispatch: A case study of illinois," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 1, pp. 250–261, 2013.

[11] J. R. Andrade, J. Filipe, M. Reis, and R. J. Bessa, "Probabilistic price forecasting for day-ahead and intraday markets: Beyond the statistical model," *Sustainability*, vol. 9, no. 11, p. 1990, 2017.

[12] 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.

[13] D. van der Meer, J. Widén, and J. Munkhammar, "Review on probabilistic forecasting of photovoltaic power production and electricity consumption," *Renewable and Sustainable Energy Reviews*, 2017.

[14] C.-M. Huang, Y.-C. Huang, K.-Y. Huang, S.-J. Chen, and S.-P. Yang, "Deterministic and probabilistic wind power forecasting using a hybrid method," in *Industrial Technology (ICIT), 2017 IEEE International Conference on*. IEEE, 2017, pp. 400–405.

[15] L. B. Godfrey and M. S. Gashler, "Neural decomposition of time-series data for effective generalization," *arXiv preprint arXiv:1705.09137*, 2017.

[16] A. S. Dorvlo, "Estimating wind speed distribution," *Energy Conversion and Management*, vol. 43, no. 17, pp. 2311–2318, 2002.

[17] P. Pinson, H. A. Nielsen, J. K. Møller, H. Madsen, and G. N. Kariniotakis, "Non-parametric probabilistic forecasts of wind power: required properties and evaluation," *Wind Energy*, vol. 10, no. 6, pp. 497–516, 2007.

[18] R. Koenker and G. Bassett Jr, "Regression quantiles," *Econometrica: journal of the Econometric Society*, pp. 33–50, 1978.

[19] R. Koenker, *Quantile regression*. Cambridge university press, 2005, no. 38.

[20] Y. Grushka-Cockayne, K. C. Lichtendahl, V. R. R. Jose, and R. L. Winkler, "Quantile evaluation, sensitivity to bracketing, and sharing business payoffs," 2016.

[21] G. Parascandolo, H. Huttunen, and T. Virtanen, "Taming the waves: sine as activation function in deep neural networks," 2016.

[22] A. Silvescu, "A new kind of neural networks," 1997.

[23] ——, "Fourier neural networks," in *Neural Networks, 1999. IJCNN'99. International Joint Conference on*, vol. 1. IEEE, 1999, pp. 488–491.

[24] K.-i. Minami, H. Nakajima, and T. Toyoshima, "Real-time discrimination of ventricular tachyarrhythmia with fourier-transform neural network," *IEEE transactions on Biomedical Engineering*, vol. 46, no. 2, pp. 179–185, 1999.

[25] L. Mingo, L. Aslanyan, J. Castellanos, M. Diaz, and V. Riazanov, "Fourier neural networks: An approach with sinusoidal activation functions," 2004.

[26] H. Tan, "Fourier neural networks and generalized single hidden layer networks in aircraft engine fault diagnostics," *Journal of engineering for gas turbines and power*, vol. 128, no. 4, pp. 773–782, 2006.

[27] K. E. Germec, "Fourier neural networks for real-time harmonic analysis," in *Signal Processing and Communications Applications Conference, 2009. SIU 2009. IEEE 17th*. IEEE, 2009, pp. 333–336.

[28] W. Zuo, Y. Zhu, and L. Cai, “Fourier-neural-network-based learning control for a class of nonlinear systems with flexible components,” *IEEE transactions on neural networks*, vol. 20, no. 1, pp. 139–151, 2009.

[29] D. Tan, W. Chen, and H. Wang, “On the use of monte-carlo simulation and deep fourier neural network in lane departure warning,” *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 4, pp. 76–90, 2017.

[30] M. S. Gashler and S. C. Ashmore, “Training deep fourier neural networks to fit time-series data,” in *International Conference on Intelligent Computing*. Springer, 2014, pp. 48–55.

[31] S. Zheng, “Gradient descent algorithms for quantile regression with smooth approximation,” *International Journal of Machine Learning and Cybernetics*, vol. 2, no. 3, p. 191, 2011.

[32] K. Ruehl, C. Michelen, S. Kanner, M. Lawson, and Y.-H. Yu, “Preliminary verification and validation of wec-sim, an open-source wave energy converter design tool,” in *33rd International Conference on Ocean, Offshore and Arctic Engineering, OMAE, San Francisco, CA, United States (Abstract accepted)*, 2014.

[33] G. Sideratos and N. D. Hatziargyriou, “Probabilistic wind power forecasting using radial basis function neural networks,” *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 1788–1796, 2012.