

Input Modeling and Uncertainty Quantification for Improving Volatile Residential Load Forecasting

Guangrui Xie^a, Xi Chen^{a,*} and Yang Weng^b

^aGrado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061 USA

^bSchool of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85281 USA

ARTICLE INFO

Keywords:

Load forecasting
Gaussian processes
Deep neural networks
Input modeling
Uncertainty quantification
Sensitivity analysis
Renewable integration

ABSTRACT

Residential load forecasting has been playing an increasingly important role in operation and planning of power systems. Over the recent years, accurate forecasts of individual loads have become ever more challenging due to the proliferation of distributed energy resources. This paper identifies and verifies the opportunity of improving load forecasting performance by incorporating suitable input modeling and uncertainty quantification, and proposes a two-stage approach that enjoys the following features. (1) It provides input modeling and quantifies the impact of input errors, rather than neglect or mitigate the impact—a prevalent practice of existing methods. (2) It propagates the impact of input errors into the ultimate point and interval predictions for the target customer's load for improved predictive performance. (3) A variance-based global sensitivity analysis method is further proposed for input-space dimensionality reduction in both stages to enhance the computational efficiency. Numerical experiments show that the proposed two-stage approach outperforms competing load forecasting methods with respect to both point predictive accuracy and coverage ability of the predictive intervals achieved.

Nomenclature

DER	Distributed energy resource
GSA	Global sensitivity analysis
NNGP	Neural network-Gaussian process
NIGP	Noisy input Gaussian process
$\mathbf{x}_t$	Input vector of the two-stage load forecasting method at hour t
N	Number of customers in the system
$\theta_{i,j}^t$	The voltage angle difference between customer i and customer j at hour t , $i \neq j$, $i, j = 1, 2, \dots, N$
P_i^h	True customer i 's load at hour h , $i = 1, 2, \dots, N$
D_j^h	Input-output pair for training NNGP at hour h , $j = 1, 2, \dots, i-1, i+1, \dots, N$
$\Theta_{i,j}^h$	A vector comprising the values of $\theta_{i,j}^t$, $t = h - 24n_{in}, \dots, h - 1$
N_u	Number of hidden units in each layer of NNGP
n_t	Number of days from which the training points are sampled for NNGP
N^*	Number of neighboring customers whose information is included for the first-stage input prediction
n_{in}	Number of days from which the data in the input layer of NNGP are sampled

d_{in}	Dimensionality of the input layer of NNGP
d_{out}	Dimensionality of the output layer of NNGP
μ_1	Predictive mean of the input vector obtained from NNGP
V_1	Predictive variance of the input vector obtained from NNGP
$\hat{S}_{T_i}^1$	Total Sobol' index estimate for the i th input in the first stage, $i = 1, 2, \dots, 24(N-1)$
$\hat{S}_{T_j}^2$	Total Sobol' index estimate for the input $\theta_{i,j}$ in the second stage, $j = 1, 2, \dots, i-1, i+1, \dots, N$
H_ℓ	Summary statistic for quantifying the impact of ℓ th component of $(\theta_{i,1}, \dots, \theta_{i,i-1}, \theta_{i,i+1}, \dots, \theta_{i,N})^\top$ on the first-stage input prediction, $\ell = 1, 2, \dots, N-1$
n_c	Number of load observations covered by a given predictive interval on a prediction day

1. Introduction

As operating decisions including management, planning, scheduling, and load dispatching rely on load forecasting, accurate forecasts are critical to achieve reliable, secure and stable operations of electric power systems. The proliferation of distributed energy resources (DERs) has been raising the level of uncertainty in power systems steadily, rendering responsive and reliable residential load forecasting more challenging than ever.

A plethora of methods have been proposed to meet the challenges in load forecasting, which can be classified into two categories: one-stage and two-stage methods. A typical one-stage method adopts a single statistical or machine

*Corresponding author.

E-mail addresses: guanx92@vt.edu (G. Xie), xchen6@vt.edu (X. Chen), yang.weng@asu.edu (Y. Weng)

ORCID(s): 0000-0001-7965-9198 (X. Chen)

learning model for load forecasting. Various models have been considered in the literature to this end, including, but not limited to, multiple linear regression (MLR), autoregressive integrated moving average (ARIMA), support vector regression (SVR), neural network (NN), and Gaussian process (GP). To give a few examples, an MLR model with polynomial terms built on temperature data was used for load forecasting in [1]. In [2], the authors adopted an MLR model with a high interpretability to investigate the causality of the consumption of electric energy. In [3], an ARIMA model with wavelet multi-resolution analysis was proposed for short-term load forecasting. In [4], the authors proposed an integrated non-seasonal and seasonal sliding window-based ARIMA model combined with the online information network technique for load prediction. In [5], an incremental learning-based SVR model using batch arriving and large datasets was proposed for load forecasting. In [6], the authors constructed an SVR model using temperature and solar flux as input variables based on sensor data for predicting energy consumption in residential buildings. In [7], a hybrid model of convolutional NN for extracting the local trends and long short-term memory NN for learning the relationship between time steps was proposed for load forecasting. In [8], the authors proposed a hybrid model of convolutional NN and gated recurrent unit NN for extracting the feature vector from high-dimensional data and from time sequence data for short-term load forecasting. In [9], a GP model with a task-specific covariance kernel incorporating seasonal and weather information was proposed for electricity demand forecasting. In [10], the authors adopted a GP model incorporating physical insights about load data characteristics to improve the load predictive accuracy achieved.

The aforementioned methods have respective advantages and disadvantages. MLR models are easy to implement and enjoy a high interpretability, but are incapable of capturing complex load patterns if not used in conjunction with other sophisticated techniques such as SVR and NN. ARIMA models can fit stationary time series well, but do not perform adequately when load series exhibit nonstationarity. SVR and NN models are known for their high predictive accuracy when modeling complex nonlinear input-output relationships, but require a high computational cost for their model training. GP models stand out from the one-stage methods: they are highly flexible to capture complex load patterns; more importantly, they can provide not only accurate point predictions but also natural interval predictions that cover the true load observations with a desirable high probability. The computational cost of GP models, however, grows rapidly with the size of the training dataset. The interested reader is referred to Table 3 in Appendix A for a summary of representative one-stage methods.

Two-stage methods perform load forecasting in two stages, as the name implies. The first stage is typically set up for processing input variables or model parameters, e.g., to select important inputs from a large candidate set or to obtain accurate estimates of input features, while the second stage carries out load forecasting using a prediction model (such

as MLR, SVR, or GP) based on the processed inputs. For example, in [11], the authors proposed to filter the electricity load signal by some feature selection technique for selecting appropriate candidate inputs to the forecast engine in the second stage for load forecasting. In [12], the first stage of the proposed approach performed the next-day average load forecasting and the second stage used the forecasts as inputs for the next-day hourly load forecasting. Two-stage methods are known for their enhanced predictive performance thanks to the first-stage processing. Table 3 in Appendix A provides a detailed account of representative two-stage load forecasting methods.

Despite many successful applications reported, existing one-stage and two-stage methods suffer from one common drawback: they fail to thoroughly address the impact of input errors when performing load predictions. Input errors arise from the use of the estimates of relevant input features for load forecasting. To achieve a high predictive accuracy, all factors affecting the load behavior can be considered as inputs for load forecasting models; these include future weather conditions (e.g., temperature, humidity, dew point, and cloud coverage) and future power states (e.g., voltages and voltage angles), etc. As the future values of the relevant inputs are unknown at the time of prediction, they must be predicted as well. The predicted input values are inevitably susceptible to errors. Neglecting input errors can result in a false sense of confidence in the predictive performance achieved by a model that fails to account for the impact. Fig. 1 provides a simple example to illustrate the importance of accounting for the impact of input errors when performing point and interval predictions via GP modeling. The input-output relationship of interest is $f(x) = \sin(12x)$ for $x \in (0, 1)$. Suppose that one is interested in making predictions at $x \in [0.7, 1]$ based on a training dataset consisted of input-output pairs obtained at 140 randomly sampled input values from $[0, 0.7]$. Specifically, a random output $Y(x)$ is generated at each equispaced x on $[0, 0.7]$ according to $Y(x) = f(x + \epsilon_x) + \epsilon_y$, where the input x is perturbed by a normally distributed input error ϵ_x with mean 0 and standard deviation 0.05; and the term ϵ_y denotes a normally distributed output error with mean 0 and standard deviation 0.2. The upper panel of Fig. 1 shows the point and interval predictions obtained by standard GP which does not account for the impact of input errors. The lower panel shows counterpart results obtained by noisy input GP (to be introduced shortly) which can effectively tackle the impact of input errors. The example manifests that noisy input GP outperforms standard GP by providing more accurate point predictions as well as interval predictions with a much higher coverage capability.

Although two-stage methods are superior to one-stage methods in terms of predictive performance; these existing methods still do not address the impact of input errors adequately. It is worth noting that some existing two-stage methods indeed aim at obtaining more accurate input estimates in the first stage by filtering out input errors via Kalman filter (KF). For instance, a two-stage method was proposed by [13]. The first stage adopted state-space models with ex-

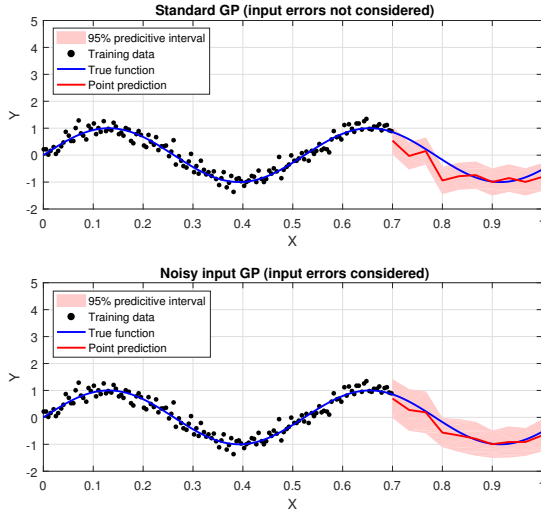


Figure 1: An illustration of the importance of accounting for the impact of input errors when performing point and interval predictions.

tended KF to model the load structure, which was used as the input to MLR models in the second stage to enhance the load forecast accuracy achieved. However, the use of KF requires the knowledge of specific system dynamics (as noted by [14]), which is typically unavailable when performing load forecasting. These facts suggest that compared to mitigating the impact of input errors, a more thorough solution is to properly quantify and propagate the impact into the point and interval load predictions via suitable methods.

This work proposes a two-stage load forecasting approach featuring input modeling and uncertainty quantification for enhanced predictive performance. The first stage of the approach performs input modeling and quantifies the input uncertainty; the second stage performs load forecasting with the input uncertainty properly propagated into the prediction results. The proposed two-stage approach is evaluated on various IEEE distribution systems that are simulated using real datasets and is found to outperform competing one-stage and two-stage methods in terms of both point and interval predictions. The major contributions of this paper can be summarized as follows:

- A novel hybrid model of NN and GP (NNGP) in the first stage to obtain accurate point predictions for future input values together with interval predictions that quantify the input uncertainty.
- A noisy input GP model (NIGP) in the second stage to produce the ultimate point and interval load predictions based on the input predictions with the input uncertainty fully taken into account.
- A model-free global sensitivity analysis (GSA) method for dynamic input feature selection to enhance the computational efficiency.

- Concrete implementation details of the two-stage approach and the GSA method on specific test cases.

The remainder of the paper is organized as follows. Section 2 elaborates on the proposed two-stage approach. Section 3 presents the GSA method that aims at enhancing the computational efficiency. Section 4 presents numerical experiments for comparing the proposed two-stage approach with competing forecasting methods. Section 5 concludes the paper.

2. Load Forecasting with Input Modeling and Uncertainty Quantification

This section elaborates on the proposed two-stage load forecasting approach and explains how it can thoroughly address the issue of input errors by quantifying and propagating their impact into the ultimate load forecasts. Notice that this work focuses on hourly load forecasting; nevertheless, other forecast resolutions can be easily adopted as needed.

For ease of exposition, define the input features to be used in the second-stage model first. Consider predicting customer or bus i 's load at hour t in a distribution grid comprising N customers by using the following vector of input features:

$$\mathbf{x}_t = \left(\theta_{i,1}^t, \dots, \theta_{i,i-1}^t, \theta_{i,i+1}^t, \theta_{i,i+2}^t, \dots, \theta_{i,N}^t \right)^\top, \quad (1)$$

where $\theta_{i,k}$ denotes the difference between the voltage angles associated with the node voltages V_i and V_k of customers i and k , and $\top$ is the transpose operator. Such a choice was first adopted by the integrated GP framework (IGP) proposed in [15], which stems from the well-known real power balance equation for single-phase distribution systems:

$$0 = -P_i + \sum_{k=1}^N |V_i||V_k|(G_{ik} \cos \theta_{i,k} + B_{ik} \sin \theta_{i,k}), \quad (2)$$

where P_i denotes the net active power injected at bus i ; $|V_i|$ denotes the voltage magnitude of bus i ; G_{ik} and B_{ik} are respectively the real and imaginary parts of the (i, k) th component of the bus admittance matrix, $i, k = 1, 2, \dots, N$. Equation (2) implies that a given target bus' load is strongly related to the voltage angle differences between other buses and itself.

For utility practice, distribution grids for buildings and residential areas are typically unbalanced multi-phase systems [16]. In this case, more sophisticated power flow analysis is required. Let a , b , and c denote the three phases of a distribution grid, respectively. The vector $\mathbf{V}_i = (V_i^a, V_i^b, V_i^c)^\top \in \mathbb{C}^3$ denotes the nodal voltages at bus i , where V_i^ϕ denotes the line-to-ground complex voltage on phase ϕ . Similarly, $\mathbf{I}_i = (I_i^a, I_i^b, I_i^c)^\top \in \mathbb{C}^3$ denotes the vector of current injections at bus i ; $\mathbf{S}_i = (S_i^a, S_i^b, S_i^c)^\top \in \mathbb{C}^3$ denotes the vector of injected complex powers at bus i ; $\mathbf{P}_i = (P_i^a, P_i^b, P_i^c)^\top \in \mathbb{C}^3$ denotes the vector of real powers at bus i ; and $\mathbf{Q}_i = (Q_i^a, Q_i^b, Q_i^c)^\top \in \mathbb{C}^3$ denotes the vector of reactive powers at bus i . If a bus is only connected with one or two phases,

the quantities of the missing phase are zeros. For example, if bus i does not have phase c , then V_i^c , I_i^c , S_i^c , P_i^c , and Q_i^c are zeros. If bus i and bus k are connected, the relationship between their nodal voltages and currents can be given as follows:

$$\begin{pmatrix} \mathbf{I}_i \\ \mathbf{I}_k \end{pmatrix} = \begin{pmatrix} \mathbf{Y}_{ik} + \frac{1}{2}\mathbf{B}_{i,\text{shunt}} & -\mathbf{Y}_{ik} \\ -\mathbf{Y}_{ik} & \mathbf{Y}_{ik} + \frac{1}{2}\mathbf{B}_{i,\text{shunt}} \end{pmatrix} \begin{pmatrix} \mathbf{V}_i \\ \mathbf{V}_k \end{pmatrix}, \quad (3)$$

where $\mathbf{Y}_{ik} \in \mathbb{C}^3$ denotes the admittance submatrix between buses i and k and $\mathbf{B}_{i,\text{shunt}} \in \mathbb{C}^3$ denotes the shunt capacitance at bus i . In a multi-phase system, $\mathbf{Y}_{ik}$ is not diagonal. The voltages at different phases are coupled. If buses i and k are not connected, $\mathbf{Y}_{ik} = \mathbf{0}$. The relationship between complex power, voltage, and current at bus i can be given as

$$\mathbf{S}_i = \mathbf{P}_i + j\mathbf{Q}_i = \mathbf{V}_i \circ \mathbf{I}_i^*, \quad (4)$$

where $j = \sqrt{-1}$ and $\circ$ denotes the Hadamard product operator which performs entrywise multiplication [17]. Notice that a strong coupling between real powers and voltage angle differences is embedded in (3) and (4).

The aforementioned power flow models for single-phase and multi-phase systems manifest that a strong coupling exists between load and voltage angle; the literature confirms this insight and further reveals that the coupling between load and voltage magnitude, however, is rather weak [18]. Voltage angle data can be conveniently collected by smart meters and have been successfully used as inputs for load forecasting [19]. Therefore, this work adopts the input features as specified in (1) for load prediction and demonstrates how to thoroughly address the impact of input errors on load forecasts via the proposed two-stage approach. Other relevant input variables useful for load forecasting (e.g., temperature and precipitation estimates) can be easily incorporated into the proposed approach, but the key issue to address above all is the impact of errors in the inputs adopted. For ease of exposition, this work focuses on load forecasting for single-phase distribution systems; the proposed approach can be easily generalized to work for multi-phase systems and achieve satisfactory predictive performance, as will be shown in Section 4.

2.1. The First Stage: Input Modeling and Error Quantification

This section first introduces the first-stage model adopted for input modeling, namely, neural network–Gaussian process (NNGP), and then details its implementation for input prediction and error quantification.

2.1.1. Model Choice

Consider predicting target customer i 's hour-ahead load P_i^{t+1} at hour t , the second-stage model in our two-stage approach requires an input vector $\mathbf{x}_{t+1}$ in the form of (1) for load prediction. Since the input values in $\mathbf{x}_{t+1}$ are unknown to us at hour t , the first-stage model is adopted to perform hour-ahead input prediction. To fully quantify the input prediction errors, the first-stage model must provide an interval prediction as well as a point prediction. GP becomes a

natural choice to this end thanks to its inherent predictive uncertainty measure. However, GP is known for its incompetent extrapolation performance [20], hence unsuitable as the first-stage model for predicting future input values. In contrast, NNs are known for their ability to capture complex output patterns and their outstanding extrapolation performance. However, NNs do not come with a predictive uncertainty measure. This work proposes to use the neural network–Gaussian process (NNGP) model recently proposed in [21] to accomplish both tasks simultaneously. The underlying idea of NNGP is to place prior distributions on the weight and bias parameters of a deep neural network (DNN) with infinite width and derive its equivalent GP formulation. By using the equivalent GP for prediction, NNGP can combine the strengths of NN and GP, providing a highly accurate point prediction as well as an informative interval prediction for future input values.

The rationale behind NNGP is briefly described next. Consider a fully-connected DNN with L hidden layers ($L \geq 2$), where the numbers of hidden units in each layer, N_u , are equal. Denote the pointwise nonlinearity function by Φ . Let $\mathbf{x} \in \mathbb{R}^{d_{in}}$ and $\mathbf{y} \in \mathbb{R}^{d_{out}}$ denote the input and output vectors of the network, respectively. Consider the i th hidden unit in the l th layer, and denote the post-nonlinearity and the post-affine transformation by a_i^l and z_i^l , respectively. It follows that $\mathbf{y} = \left(z_1^L(\mathbf{x}), z_2^L(\mathbf{x}), \dots, z_{d_{out}}^L(\mathbf{x}) \right)^\top$. Denote the weight and bias parameters in the l th layer by $\mathbf{W}_{ij}^l$ and b_i^l ; we place normal prior distributions with zero means and variances σ_w^2/N_u and σ_b^2 respectively on $\mathbf{W}_{ij}^l$ and b_i^l .

To derive an equivalent GP for a generic DNN at hand, we first show how to do so for a one-layer NN with infinite width. Consider the output of the i th hidden unit of the first layer in the NN, it follows that

$$\begin{aligned} z_i^1(\mathbf{x}) &= b_i^1 + \sum_{j=1}^{N_u} \mathbf{W}_{ij}^1 a_j^1(\mathbf{x}), \\ \text{with } a_j^1(\mathbf{x}) &= \Phi\left(b_j^0 + \sum_{k=1}^{d_{in}} \mathbf{W}_{jk}^0 x_k\right), \end{aligned} \quad (5)$$

where the superscript “0” denotes the input layer of the NN, and x_k denotes the k th component of the input vector $\mathbf{x}$. As $z_i^1(\mathbf{x})$ is the sum of independent and identically distributed terms, the standard Central Limit Theorem indicates that $z_i^1(\mathbf{x})$ becomes normally distributed as N_u approaches infinity. Hence, given any input vectors $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^n$, $\mathbf{z}_i^1 = (z_i^1(\mathbf{x}^1), z_i^1(\mathbf{x}^2), \dots, z_i^1(\mathbf{x}^n))^\top$ follows a multivariate normal distribution. That is, $\mathbf{z}_i^1 \sim \mathcal{N}(\mu^1, K^1)$, where K^1 denotes the $n \times n$ covariance matrix for the first layer, with its (g, h) th component given by $K^1(g, h) = \text{Cov}(z_i^1(\mathbf{x}^g), z_i^1(\mathbf{x}^h))$, $g, h = 1, 2, \dots, n$. Since the parameters $\mathbf{W}_{ij}^1$'s and b_i^1 's have zero means, $\mu^1(\mathbf{x}) = \mathbb{E}[z_i^1(\mathbf{x})] = 0$ for any $\mathbf{x} \in \mathbb{R}^{d_{in}}$. In particular, given any two inputs $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^{d_{in}}$, it follows that

$$K^1(\mathbf{x}, \mathbf{x}') = \mathbb{E}[z_i^1(\mathbf{x})z_i^1(\mathbf{x}')] = \sigma_b^2 + \sigma_w^2 \mathbb{E}[a_i^1(\mathbf{x})a_i^1(\mathbf{x}')]. \quad (6)$$

Therefore, each component of the output vector of a one-layer NN with infinite width is equivalent to a GP with a covariance function specified by (6).

The aforementioned result can be generalized to NNs with more than one layer in a recursive manner by utilizing the relationship between a_j^l and z_j^{l-1} , i.e., $a_j^l(\mathbf{x}) = \Phi(z_j^{l-1}(\mathbf{x}))$. The resulting covariance function in the L th layer, $K^L(\mathbf{x}, \mathbf{x}')$, hence serves as the covariance kernel of the equivalent GP for a DNN with L hidden layers. The interested reader is referred to Appendix A of [22] for a detailed derivation.

2.1.2. Input Prediction and Error Quantification via Neural Network–Gaussian Process

The first stage of the proposed approach predicts the hour-ahead input vector component by component via NNGP. For predicting each component, the first stage utilizes not only their own values observed in the past but also the observed values of other components.

Consider predicting the input $\theta_{i,j}$ at hour h^* , for $j = 1, 2, \dots, i-1, i+1, \dots, N$ (see the definition given in (1)). One can first construct a training set for NNGP as $\{D_j^{h^*-24n_{t_1}}, D_j^{h^*-24n_{t_1}+1}, \dots, D_j^{h^*-1}\}$, where D_j^h denotes an input-output pair (X_j^h, Y_j^h) for $h = h^* - 24n_{t_1}, h^* - 24n_{t_1} + 1, \dots, h^* - 1$, and n_{t_1} denotes the number of days from which the training points are sampled. The output layer Y_j^h is in fact $\theta_{i,j}^h$, namely, the voltage angle difference between customer i and customer j at hour h . Taking into account the correlations between different customers (recall (1)-(2) and discussions therein), set $X_j^h = (\Theta_{i,j}^{h\top}, \Theta_{i,j+1}^{h\top}, \dots, \Theta_{i,j+N^*}^{h\top})^\top$, where $\Theta_{i,j}^h = (\theta_{i,j}^{h-24n_{in}}, \dots, \theta_{i,j}^{h-1})^\top$. The indices $j^1, j^2, \dots, j^{N^*}$ are selected from the set $\{1, 2, \dots, N\}$ excluding i and j , and n_{in} denotes the number of days from which the data in the input layer are sampled.

Upon constructing the training set, an NNGP model can be constructed using the covariance kernel obtained following the steps as given in Section 2.1.1. The predictive mean and variance of each component in the hour-ahead input vector can be obtained via NNGP in the same manner as using a standard GP. Hence, the predictive mean of the input vector,

$$\mu_1 = (\mathbb{E}[\theta_{i,1}], \dots, \mathbb{E}[\theta_{i,i-1}], \mathbb{E}[\theta_{i,i+1}], \dots, \mathbb{E}[\theta_{i,N}])^\top, \quad (7)$$

and predictive variance of the input vector,

$$V_1 = (\mathbb{V}[\theta_{i,1}], \dots, \mathbb{V}[\theta_{i,i-1}], \mathbb{V}[\theta_{i,i+1}], \dots, \mathbb{V}[\theta_{i,N}])^\top, \quad (8)$$

are readily available, which can then be used in the second-stage model for making ultimate load forecasts. The predictive mean μ_1 provides a point prediction for the input vector, which in conjunction with the predictive variance V_1 can be used to construct an interval prediction for each component of the input vector. Consider obtaining an interval prediction for the component $\theta_{i,j}$ ($j \neq i, j = 1, 2, \dots, N$) for an example. The upper and lower bounds of the commonly adopted 95% predictive interval for $\theta_{i,j}$ can be given by $\mathbb{E}[\theta_{i,j}] \pm 1.96\sqrt{\mathbb{V}[\theta_{i,j}]}$.

The choice of the parameters for NNGP (i.e., $n_{t_1}, n_{in}, \sigma_b, \sigma_w$, and the number of hidden layers L) greatly impacts the predictive accuracy achieved. Appropriate parameter values can be obtained by a cross-validation procedure; the interested reader is referred to [21] for details.

2.2. The Second Stage: Load Forecasting with Input Error Propagation and Uncertainty Quantification

This section first introduces the second-stage model adopted for load forecasting, noisy input Gaussian process (NIGP), which can fully address the impact of input errors, and then proceeds to describe the implementation of NIGP.

2.2.1. Model Choice

The second stage of the proposed approach aims at appropriately propagating the impact of input errors assessed in the first stage into the ultimate load forecasting results, reflecting it in both point and interval load predictions. The noisy input Gaussian process (NIGP) model proposed in [23] is arguably an adequate choice as the second-stage model. Compared to standard GP models, NIGP can propagate the impact of input errors into both point and interval predictions, providing a thorough solution to the issue of input errors in load forecasting.

The rationale behind NIGP is briefly described next. Assume that the observed output $y \in \mathbb{R}$ is a noisy measurement of an actual output $\tilde{y}$, that is,

$$y = \tilde{y} + \varepsilon_y, \quad (9)$$

where the output noise is assumed to be normally distributed, i.e., $\varepsilon_y \sim \mathcal{N}(0, \sigma_y^2)$, with σ_y^2 denoting the output noise variance. Assume that the observed input vector $\mathbf{x} \in \mathbb{R}^d$ can be modeled as the actual input vector $\tilde{\mathbf{x}}$ corrupted by some input noise:

$$\mathbf{x} = \tilde{\mathbf{x}} + \varepsilon_x, \quad (10)$$

where the input noise vector is assumed to be normally distributed, i.e., $\varepsilon_x \sim \mathcal{N}(0, \Sigma_x)$, with Σ_x denoting the $d \times d$ input noise covariance matrix. The relationship between the observed input and output follows immediately from (9) and (10):

$$y = f(\mathbf{x} - \varepsilon_x) + \varepsilon_y, \quad (11)$$

where $f : \mathbb{R}^d \rightarrow \mathbb{R}$ denotes the target input-output relationship to estimate. Suppose that $f(\cdot)$ can be modeled as a GP with some covariance kernel. The first-order Taylor expansion of $f(\mathbf{x} - \varepsilon_x)$ yields

$$f(\mathbf{x} - \varepsilon_x) = f(\mathbf{x}) - \varepsilon_x^\top \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} + \dots, \quad (12)$$

where $\partial f(\cdot)/\partial \mathbf{x}$ denotes the partial derivative process of $f(\cdot)$, which is also a GP [24]. Given that the distribution of the product of two normal random vectors $\varepsilon_x^\top$ and $\partial f(\mathbf{x})/\partial \mathbf{x}$ is analytically intractable, one can approximate $\partial f(\mathbf{x})/\partial \mathbf{x}$ by

the gradient of the posterior mean of $f(\mathbf{x})$; denote it by $\partial \bar{\mathbf{f}}$. It then follows from (11) and (12) that

$$y \approx f(\mathbf{x}) - \epsilon_x^\top \partial \bar{\mathbf{f}} + \epsilon_y, \quad (13)$$

which implies that the observed output y is approximately normally distributed, i.e., $P(y|f) = \mathcal{N}(f(\mathbf{x}), \sigma_y^2 + \partial \bar{\mathbf{f}}^\top \Sigma_x \partial \bar{\mathbf{f}})$.

Now suppose that the training set for NIGP comprises N observed input-output pairs, $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$. Denote the $N \times d$ input matrix by $\mathbf{X}$ and the N -dimensional output vector by $\mathbf{Y}$. Let $\mathbf{C}$ denote the $N \times N$ covariance matrix obtained by evaluating the covariance kernel at $\mathbf{X}$, and denote the gradient of the posterior means at the N training inputs by $\Delta \bar{\mathbf{f}}$, which is an $N \times d$ matrix. When the popular squared exponential covariance kernel is used, the covariance between any two inputs $\mathbf{x}_i$ and $\mathbf{x}_j$ is given by

$$\mathbf{C}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)^\top \Lambda^{-1}(\mathbf{x}_i - \mathbf{x}_j)\right), \quad (14)$$

where σ_f^2 represents the process variance, and Λ denotes the $d \times d$ diagonal matrix with its main diagonal elements being the length-scale hyper-parameters.

The expressions for the predictive mean and variance of NIGP can be derived when the test input is either deterministic or stochastic. Denote the observed test input by $\mathbf{x}_*$ and the actual test input by $\tilde{\mathbf{x}}_*$. In the deterministic case, $\mathbf{x}_* = \tilde{\mathbf{x}}_*$. It follows that

$$\begin{aligned} \mathbb{E}[f(\mathbf{x}_*)|\mathbf{X}, \mathbf{Y}] &= \mathbf{C}(\tilde{\mathbf{x}}_*, \mathbf{X})[\mathbf{C}(\mathbf{X}, \mathbf{X}) + \sigma_y^2 \mathbf{I}_N \\ &\quad + \text{diag}(\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top)]^{-1} \mathbf{Y}, \end{aligned} \quad (15)$$

$$\begin{aligned} \mathbb{V}[f(\mathbf{x}_*)|\mathbf{X}, \mathbf{Y}] &= \mathbf{C}(\tilde{\mathbf{x}}_*, \tilde{\mathbf{x}}_*) - \mathbf{C}(\tilde{\mathbf{x}}_*, \mathbf{X})[\mathbf{C}(\mathbf{X}, \mathbf{X}) + \sigma_y^2 \mathbf{I}_N \\ &\quad + \text{diag}(\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top)]^{-1} \mathbf{C}(\mathbf{X}, \tilde{\mathbf{x}}_*), \end{aligned} \quad (16)$$

where $\text{diag}(\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top)$ denotes the diagonal of the matrix $\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top$, and $\mathbf{I}_N$ denotes the $N \times N$ identity matrix. Notice that $\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top$ is a correction term added to the covariance matrix of a standard GP model to reflect the uncertainty associated with the inputs in $\mathbf{X}$.

In the stochastic case, assume that the observed test input $\mathbf{x}_*$ is multivariate normally distributed, i.e., $\mathbf{x}_* \sim \mathcal{N}(\tilde{\mathbf{x}}_*, \Sigma_x)$. It follows that the predictive mean of $f(\mathbf{x}_*)$ can be given as

$$\mathbb{E}[f(\mathbf{x}_*)] = \left(\left[\mathbf{C} + \sigma_y^2 \mathbf{I}_N + \text{diag}(\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top) \right]^{-1} \mathbf{Y} \right)^\top \mathbf{q}. \quad (17)$$

Compared to the predictive mean given in (15), $\mathbf{q}$ is the counterpart of $\mathbf{C}(\tilde{\mathbf{x}}_*, \mathbf{X})$ in the stochastic case, which is given by

$$\mathbf{q} = \int \mathbf{C}(\mathbf{x}_*, \mathbf{X}) p(\mathbf{x}_*|\tilde{\mathbf{x}}_*, \Sigma_x) d\mathbf{x}_*, \quad (18)$$

where $p(\mathbf{x}_*|\tilde{\mathbf{x}}_*, \Sigma_x)$ is the posterior of $\mathbf{x}_*$. When the squared exponential covariance kernel in (14) is adopted, $\mathbf{q}$ is an $N \times 1$ vector whose i th component is given by

$$q_i = \sigma_f^2 |\Sigma_x \Lambda^{-1} + \mathbf{I}_d|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \tilde{\mathbf{x}}_*)^\top (\Sigma_x + \Lambda)^{-1}(\mathbf{x}_i - \tilde{\mathbf{x}}_*)\right), \quad i = 1, 2, \dots, N, \quad (19)$$

where $\mathbf{I}_d$ denotes the $d \times d$ identity matrix. Utilizing the law of total variance, the predictive variance of $f(\mathbf{x}_*)$ follows as

$$\begin{aligned} \mathbb{V}[f(\mathbf{x}_*)] &= \sigma_f^2 + \alpha^\top \mathbf{Q} \alpha - (\mathbb{E}[f(\mathbf{x}_*)])^2 \\ &\quad - \text{tr}([\mathbf{C} + \sigma_y^2 \mathbf{I}_N + \text{diag}(\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top)]^{-1} \mathbf{Q}) \end{aligned} \quad (20)$$

where $\text{tr}(\mathbf{M})$ denotes the trace of matrix $\mathbf{M}$, and $\mathbf{Q}$ is an $N \times N$ matrix whose (i, j) th component is given by

$$\begin{aligned} \mathbf{Q}_{ij} &= \frac{\mathbf{C}(\mathbf{x}_i, \tilde{\mathbf{x}}_*) \mathbf{C}(\mathbf{x}_j, \tilde{\mathbf{x}}_*)}{|\Sigma_x \Lambda^{-1} + \mathbf{I}_d|^{\frac{1}{2}}} \exp\left((\mathbf{z} - \tilde{\mathbf{x}}_*)^\top (\Lambda + \frac{1}{2} \Lambda \Sigma_x^{-1} \Lambda)^{-1} (\mathbf{z} - \tilde{\mathbf{x}}_*)\right), \quad \text{with } \mathbf{z} = (\mathbf{x}_i + \mathbf{x}_j)/2, \end{aligned} \quad (21)$$

and $\alpha = (\mathbf{C} + \sigma_y^2 \mathbf{I}_N + \text{diag}(\Delta \bar{\mathbf{f}} \Sigma_x \Delta \bar{\mathbf{f}}^\top))^{-1} \mathbf{Y}$. The interested reader is referred to [23] for more details on the derivation of the predictive mean and variance functions of NIGP. A MATLAB package is available for implementing NIGP [25].

It is worth noting that there exist other machine learning methods that can produce reliable point predictions based on uncertain inputs. For instance, in [26], a feed-forward inference method was proposed to propagate input uncertainty through all layers of an NN; in [27], the authors adopted a Bayesian NN framework to account for the input noise given the knowledge of the noise process model. However, to the best of our knowledge, NIGP is the only approach that can provide an adequate uncertainty measure for the ultimate prediction delivered. Moreover, by using different covariance kernels, NIGP enjoys the flexibility of capturing a wide range of customers' load patterns, which is highly valuable in real-life applications.

2.2.2. Load Forecasting via Noisy Input Gaussian Process with Input Error Propagation and Uncertainty Quantification

In the second stage, NIGP can be conveniently implemented for making ultimate load forecasts. As NIGP is essentially a variant of GP, its training and prediction can be performed in the same manner as using a standard GP. Consider predicting the target customer i 's hour-ahead load P_i^{h*} at hour $h^* - 1$. One can first estimate the hyper-parameters σ_f^2 , Λ and σ_y^2 via maximum likelihood estimation based on a set of training input-output pairs $\{(\mathbf{x}_h, P_i^h)\}_{h=1}^{n_{t_2}}$, where n_{t_2} denotes the number of days from which the training points are sampled, and $\mathbf{x}_h$ is in the form of (1). With the estimates of the hyper-parameters, load forecasting can be made based on the predictive mean and variance of the input vector, μ_1 and $\mathbf{V}_1$, respectively given by (7) and (8). Specifically, set $\Sigma_1 = \text{diag}(\mathbf{V}_1)$, a diagonal matrix whose main diagonal entries are given by the components of $\mathbf{V}_1$. Then the predictive mean and variance for the target customer's hour-ahead load P_i^{h*} can be obtained by replacing $\tilde{\mathbf{x}}_*$ and Σ_x in (17) and (20) respectively by μ_1 and Σ_1 . The predictive mean $\mathbb{E}[f(\mathbf{x}_*)]$ given in (17) can be used as a point prediction for P_i^{h*} , which in conjunction with the predictive variance $\mathbb{V}[f(\mathbf{x}_*)]$ given in (20) can be utilized to construct an interval prediction for P_i^{h*} . Specifically, the upper and lower bounds of the com-

monly adopted 95% predictive interval for P_i^{h*} are given by $\mathbb{E}[f(\mathbf{x}_*)] \pm 1.96\sqrt{\mathbb{V}[f(\mathbf{x}_*)]}$.

3. Improving Computational Efficiency via Global Sensitivity Analysis

This section aims at enhancing the computational efficiency of the proposed two-stage load forecasting approach via global sensitivity analysis (GSA). Due to the use of spatial information (i.e. voltage angle difference) for load forecasting, the input-space dimensionality can increase dramatically with the number of customers in the power grid, resulting in a high computational cost. To reduce the input-space dimensionality, one can consider retaining only a few most important input dimensions for load forecasting. Although many feature selection techniques are available, e.g., filter methods, wrapper methods, and embedded methods (for a review, see [28, 29]), most of them are parametric approaches and hence pose restrictive assumptions on the underlying input-output relationship. This work proposes to use the variance-based GSA method, which is model-free and hence works well without assuming any particular input-output relationship [30]. GSA can be applied to both stages of the proposed approach to improve the computational efficiency achieved.

3.1. Variance-Based Global Sensitivity Analysis for Systems with Functional Inputs

GSA focuses on quantifying how sensitive the output is to each individual input feature and their interactions, and is useful for identifying those inputs that contribute the most to the output variability. GSA methods fall into two categories: regression-based and variance-based. The main idea of the variance-based GSA methods is to decompose the variance of the output as a sum of contributions of each input feature. One of the most widely used variance-based GSA methods is the Sobol' index method, on which a brief overview is provided next.

Consider the following model: $X \mapsto Y = f(X)$, where $f(\cdot)$ denotes the underlying input-output function, $X = (X_1, X_2, \dots, X_m)^T$ denotes the $m \times 1$ vector of input features, and Y represents the output. The first-order Sobol' index of X_i is defined as

$$S_i = \frac{\mathbb{V}_{X_i}(\mathbb{E}_{X_{\sim i}}[Y|X_i])}{\mathbb{V}(Y)}, \quad i = 1, 2, \dots, m, \quad (22)$$

which quantifies the impact of X_i on Y only, excluding any interactions with other inputs. The total Sobol' index of X_i is defined as

$$S_{T_i} = \frac{\mathbb{E}_{X_{\sim i}}[\mathbb{V}_{X_i}(Y|X_{\sim i})]}{\mathbb{V}(Y)} = 1 - \frac{\mathbb{V}_{X_{\sim i}}(\mathbb{E}_{X_i}[Y|X_{\sim i}])}{\mathbb{V}(Y)}, \quad (23)$$

where $X_{\sim i}$ denotes the collection of all inputs except X_i . The total Sobol' index measures the total impact of X_i on Y , including higher-order impacts through interactions with other inputs. By definition, a Sobol' index takes a value in $[0, 1]$.

The larger an index value is, the greater impact the associated input has on the output.

A majority of existing methods proposed for Sobol' index estimation focus on tackling spatial inputs. To give a few examples, different approaches for calculating total Sobol' indices for spatial inputs were compared in [31]. In [32], the authors proposed an innovative method based on the functional decomposition of the output to obtain a spatial map of the Sobol' indices. In [33], a novel copula-based approach for estimating Sobol' indices for models with dependent spatial input variables was presented. In the context of load forecasting, however, methods capable of handling functional inputs (i.e., inputs that vary with time) are required. GSA methods that tackle systems with functional inputs are relatively scarce. In [34], the authors proposed an effective Sobol' index estimation method for GSA of dynamic systems, which is suitable to be applied to the first- and second-stage analyses to improve the computational efficiency achieved.

To estimate Sobol' indices, one can first decompose the functional inputs through a simultaneous principle component analysis as follows:

$$X = \sum_{i=1}^p \beta_i \gamma_i, \quad (24)$$

where $\gamma = \{\gamma_1, \gamma_2, \dots, \gamma_p\}$ denotes the collection of basis functions such as B-splines and wavelets, and $\beta = \{\beta_1, \beta_2, \dots, \beta_p\}$ denotes the set of corresponding coefficients, with p denoting the number of basis functions used. Then the joint distribution of the coefficients in β can be approximated using a Gaussian mixture model. Subsequently, one can generate Monte Carlo (MC) samples of the functional inputs based on (24) by sampling from the approximated distribution of β . Given a sample of input features, the corresponding outputs Y can be obtained using an approximated input-output function $\hat{f}$. The first-order and total Sobol' indices can then be estimated via MC sampling. Taking into account the potential interactions between the components of the input vector, one can adopt the total Sobol' index as the importance measure and use the following estimator proposed in [35]:

$$\hat{S}_{T_i} = \frac{(2n)^{-1} \sum_{j=1}^n \left(\hat{f}(\mathbf{A})_j - \hat{f}(\mathbf{A}_{\mathbf{B}}^{(i)})_j \right)^2}{\mathbb{V}(\hat{f}(\mathbf{C}))}, \quad (25)$$

where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ respectively denote three independently generated $n \times m$ matrices of input samples, each row of which gives a randomly sampled input vector. In (25), $\hat{f}(\mathbf{A})$ evaluates the approximated function $\hat{f}$ at the input vector given by each row of matrix $\mathbf{A}$, and the subscript j of $\hat{f}(\mathbf{A})_j$ denotes the j th element of $\hat{f}(\mathbf{A})$. $\mathbf{A}_{\mathbf{B}}^{(i)}$ denotes the matrix whose i th column is taken from matrix $\mathbf{B}$ while the other $m-1$ columns remain from matrix $\mathbf{A}$.

3.2. Speeding Up Input Prediction via Global Sensitivity Analysis

The purpose of performing GSA in the first stage is to select a set of customer indices $\{j^1, j^2, \dots, j^{N^*}\}$ from $\{1, 2, \dots, N\}$ (see Section 2.1.2) to help with input prediction. Consider predicting the value of input $\theta_{i,j}$ at each hour on a particular day, namely, $\theta_{i,j}^{h^*}, \dots, \theta_{i,j}^{h^*+23}$. Prior to prediction, one can first construct a training set with full-dimensional inputs for the GSA purpose. Specifically, set $N^* = N - 2$ and $n_{in} = 1$ in Section 2.1.2. Hence, each input dimension corresponds to the voltage angle difference of the target customer and another customer at one of the previous 24 hours and the input-space dimensionality is $24(N - 1)$. Recall that in the first stage, the input-output relationship is approximated by NNGP, hence the first-stage GSA is carried out based on the NNGP model to obtain the total index estimates, $\hat{S}_{T_i}^1$, $i = 1, 2, \dots, 24(N - 1)$. Since each index i corresponds to a particular hour and a particular customer, one can then quantify the impact of the ℓ th component of the input vector $(\theta_{i,1}, \dots, \theta_{i,i-1}, \theta_{i,i+1}, \dots, \theta_{i,N})^\top$ by the following summary statistic:

$$H_\ell = \sum_{i=24(\ell-1)+1}^{24\ell} \hat{S}_{T_i}^1, \quad \ell = 1, 2, \dots, N - 1. \quad (26)$$

The larger the value of H_ℓ , the greater the impact of the ℓ th component. One can sort the H_ℓ 's in a non-increasing order and select the first N^* corresponding components (excluding $\Theta_{i,j}^h$) to be included in X_j^h for input prediction as detailed in Section 2.1.2. The value of N^* can be determined together with other parameters of NNGP by running a cross-validation procedure.

3.3. Speeding Up Load Forecasting via Global Sensitivity Analysis

The aim of performing GSA in the second stage is to identify and select a few important components in the input vector $(\theta_{i,1}, \dots, \theta_{i,i-1}, \theta_{i,i+1}, \dots, \theta_{i,N})^\top$ to be used in the NIGP model for predicting the target customer i 's hour-ahead load. One can carry out GSA based on the NIGP model trained in the second stage following the steps detailed in Section 3.1, and obtain the total Sobol' index estimate, $\hat{S}_{T_j}^2$, for each input $\theta_{i,j}$. The higher $\hat{S}_{T_j}^2$ is, the more important $\theta_{i,j}$ is considered for load prediction of the target customer i . Upon identifying the most important components in the input vector, only those components will be retained as inputs for future load prediction. Hence, a higher computational efficiency can be achieved.

4. Numerical Experiments

This section provides a comprehensive evaluation of the proposed two-stage load forecasting approach (referred to as NNGP-NIGP hereinafter) in comparison with representative one-stage and two-stage methods.

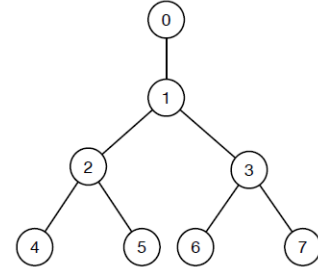


Figure 2: The topology of the 8-bus test cases. A node represents a bus, which can be single-phase or multi-phase. An edge represents a branch between two buses. Bus 0 is the substation (root).

4.1. Experimental Setup

Data preparation. Experiments are conducted extensively on single-phase and multiple-phase test systems, including the IEEE 8-bus, 24-bus, 123-bus test cases. In a single-phase test system, each bus has a single-phase load, while in a multi-phase test system, each bus has a three-phase load. As the results are similar, for the sake of brevity, this section only presents the results that cover different aspects of the proposed two-stage forecasting approach in comparison with the benchmarking methods. Fig. 2 illustrates the topology of the single-phase and three-phase 8-bus systems. To simulate highly uncertain load behaviors caused by DERs in real-life power systems, historical load profiles from Pennsylvania-New Jersey-Maryland system operator [36] in year 2014 and New York Independent System Operator [37] in year 2015 are used for simulations. Notice that in all test cases loads are given in per-unit values, with a base load of 100 kW. Fig. 3 illustrates the load observed for an arbitrarily selected customer in the single-phase 8-bus system throughout a year and the corresponding moving coefficient of variation (MCV). The data are drawn from PJM [36] and sampled hourly, and $MCV = s/\mu$, with s and μ respectively denoting the moving standard deviation and moving average of the load observed over five hours. Fig. 3 manifests that the load pattern is highly volatile as the MCV can be as high as greater than 2. Hence, load forecasting in the test cases is considerably challenging to perform. Taking into account the uncertain renewable generation behaviors of DERs, we first pre-process the hourly PV generation data over a year drawn from Renewable.ninja [38], and then subtract the pre-processed data from the load data of each bus. To obtain voltage angle values, we perform power flow analysis to generate the states of the power system hourly over a one-year time frame using the MATLAB Power System Simulation Package (MATPOWER) [39, 40], based on the processed load data.

Benchmarking methods. NNGP-NIGP is compared with three benchmarking prediction methods. The first method considered is one of the most commonly used one-stage load forecasting models, the SVR model proposed in [6]. For fairness of comparison, the same input features as given by (1) are adopted by the SVR model. The comparison be-

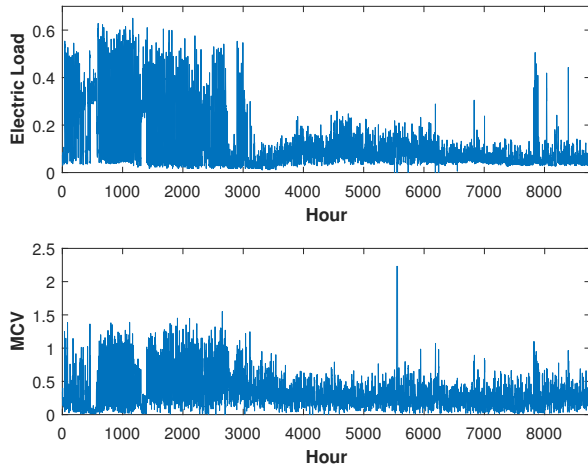


Figure 3: Hourly load of a customer in the single-phase 8-bus test case and the corresponding moving coefficient of variation (MCV) observed throughout one year. The load is given in per-unit value.

tween NNGP-NIGP and SVR helps assess if NNGP-NIGP outperforms a popular one-stage method in load forecasting. The second benchmarking method is a representative two-stage model developed recently in [15]—IGP. Recall that IGP adopts the same input features as NNGP-NIGP as specified in (1). Nonetheless, the first stage of IGP provides a point prediction for the hour-ahead input vector via k -means clustering, failing to quantify and propagate the impact of input errors into ultimate load forecasts. Without a mechanism to properly tackle the uncertain inputs, IGP constructs interval load predictions for a target customer via a heuristic approach. The comparison between NNGP-NIGP and IGP helps reveal if substantial improvements in the predictive performance can be achieved through proper input error propagation and quantification. The third benchmarking method is a two-stage model which combines Kalman filter with GP (referred to as KF-GP hereinafter). KF-GP applies Kalman filter in the first stage to filter out the input prediction errors and feeds the input prediction to a GP model for load forecasting in the second stage. The comparison between NNGP-NIGP and KF-GP helps investigate if input error propagation and quantification works more effectively than input error filtering.

Measures of predictive performance. The predictive performance of each method is evaluated by both point predictive accuracy and coverage ability of pointwise predictive intervals obtained. To evaluate point predictive accuracy, we use the mean absolute percentage error (MAPE), which is defined as

$$\text{MAPE} = \frac{1}{T} \sum_{h=1}^T \left| \frac{P_h - \hat{P}_h}{P_h} \right|, \quad (27)$$

where P_h and $\hat{P}_h$ respectively denote the load actually observed at hour h and the predicted value, and T denotes the

number of hourly predictions made. The coverage ability is assessed by the coverage probability (CP) of individual pointwise 95% predictive intervals, which is defined as

$$\text{CP} = n_c / T, \quad (28)$$

where n_c denotes the number of load observations covered by a given predictive interval on a prediction day. Given each load forecasting method under comparison, the MAPE and CP for each prediction day are obtained using $T = 24$ for all customers throughout a year in each test case considered. Notice that separate MAPE and CP values are obtained for each phase in the three-phase test cases.

Model configuration. Appropriate parameter values for the first-stage NNGP model and GSA are found via cross validation, which give the lowest average MAPE throughout a year. For the 8-bus and 24-bus test cases, the parameter setting used is $n_t = 60$, $n_{in} = 3$, $\sigma_b = 1$, $\sigma_w = 1$, and $N^* = 1$ (the number of neighboring customers whose information is included for the first-stage input prediction; see Section 3.2 for details). For the 123-bus test cases, the parameter setting used is $n_t = 90$, $n_{in} = 3$, $\sigma_b = 1$, $\sigma_w = 1$, and $N^* = 2$. Regarding the second-stage NIGP model, the only parameters that require estimation are those in the covariance kernel, which are learned via maximum likelihood estimation.

4.2. Comparison of Point and Interval Predictions

This section provides a comprehensive comparison of the proposed two-stage approach (NNGP-NIGP) with the benchmarking methods in terms of point and interval load predictions.

Implementation of the proposed two-stage approach. Consider the single-phase 8-bus test case to illustrate the proposed two-stage approach for load forecasting. Fig. 4 shows a schematic diagram of the proposed two-stage approach. Assume that one is at hour $h^* - 1$ and aims to predict customer i 's load at hour h^* . In Step 1, one can first collect historical voltage angle and load data generated by MATPOWER to construct a training dataset for NNGP as detailed in Section 2.1.2. In Step 2, one can obtain the predictive mean and variance of the input vector μ_1 and V_1 at hour h^* as given in (7) and (8) based on the NNGP model constructed. The point prediction of inputs is given by (7), and the interval prediction of inputs can be constructed as detailed in Section 2.1.2. In Step 3, μ_1 and V_1 are used by the second-stage NIGP model for ultimate load forecasting with proper input uncertainty propagation. Specifically, the point load prediction is given by the predictive mean in (17), and the load interval prediction can be constructed subsequently based on the predictive variance in (20) as detailed in Section 2.2.2. The aforementioned process can be carried out sequentially to obtain point and interval load forecasts for all customers (and for each phase in case of a multi-phase system). For the sake of brevity, the implementation details of the proposed two-stage approach for other test cases are omitted.

Summary of results. The MAPEs for the single-phase test

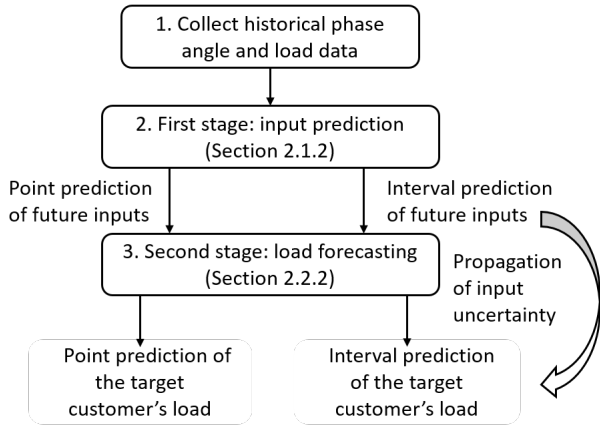


Figure 4: A schematic diagram of the proposed two-stage load forecasting approach.

cases and the three-phase 8-bus test case obtained by NNGP-NIGP and the three benchmarking methods, KF-GP, IGP, and SVR, are summarized in Fig. 5. For clarity of the figure, the results corresponding to the other three-phase test cases are omitted. It can be observed that across all test cases, NNGP-NIGP performs the best for point prediction by producing the lowest MAPEs, followed by KF-GP, IGP, and SVR. This is not surprising as NNGP-NIGP can properly propagate the impact of input errors into the ultimate point prediction for the target customer's load while the other three methods can not. Although KF-GP can mitigate the impact of input errors to some extent by filtering, which leads to better point predictive performance than IGP and SVR, KF-GP underperforms NNGP-NIGP in terms of capturing the highly nonlinear patterns of voltage angles, and subsequently the predictive performance of KF-GP is inferior to that of NNGP-NIGP. It is worth noting that the point predictive accuracy achieved by NNGP-NIGP is quite satisfactory when applied for all test cases, taking into account the highly volatile load behaviors observed in the test cases as shown in Fig. 3.

The CPs for the single-phase test cases and the three-phase 8-bus test case obtained by NNGP-NIGP, KF-GP, IGP, and SVR are summarized in Fig. 6. For clarity of the figure, the results corresponding to the other three-phase test cases are omitted. It can be seen that NNGP-NIGP produces a higher coverage probability than the other three methods across all test cases considered. To closely examine the ability of NNGP-NIGP in appropriately propagating the impact of input errors into the predictive uncertainty associated with the target customer's load forecasts, we show in Fig. 7 through Fig. 9 the 95% predictive intervals produced respectively by NNGP-NIGP and the three benchmarking models for an arbitrarily selected customer on a typical prediction day in all single-phase test cases. It is observed that the predictive intervals given by NNGP-NIGP can cover more load observations as compared to the other methods. Moreover, the curves of the predictive mean and interval bounds provided by NNGP-NIGP are smoother than those produced by the

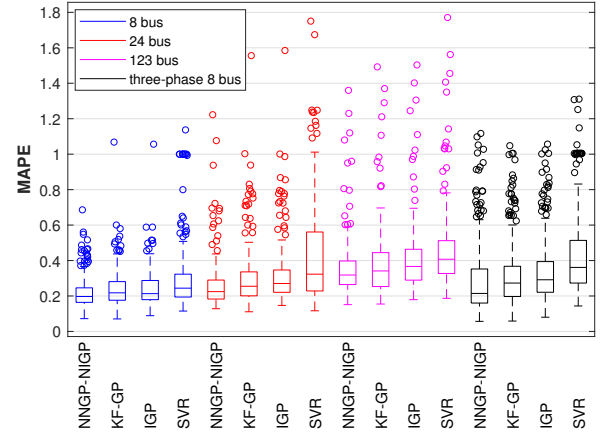


Figure 5: Boxplots of MAPEs obtained by NNGP-NIGP, KF-GP, IGP, and SVR in various test cases.

other methods; this will facilitate power systems planning and operations, as utilities can avoid making abrupt changes in their resource allocation.

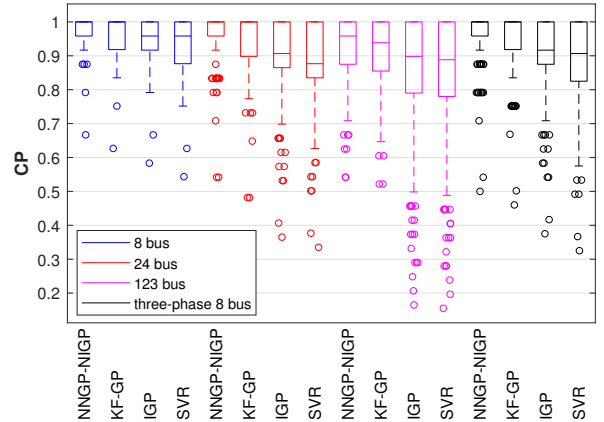


Figure 6: Boxplots of coverage probabilities obtained by NNGP-NIGP, KF-GP, IGP, and SVR in various test cases.

4.3. Applying Global Sensitivity Analysis to Improve Computational Efficiency

This section demonstrates the capability of the proposed GSA method in enhancing the computational efficiency of the proposed two-stage approach (NNGP-NIGP) for load forecasting.

Implementation. Consider the single-phase 8-bus test case to illustrate the proposed GSA method for enhancing the computational efficiency of load forecasting for customer i at a given hour h^* . To perform GSA in the first stage for reducing the computational cost associated with input modeling via NNGP, one can first calculate H_ℓ 's for $\ell = 1, 2, \dots, N -$

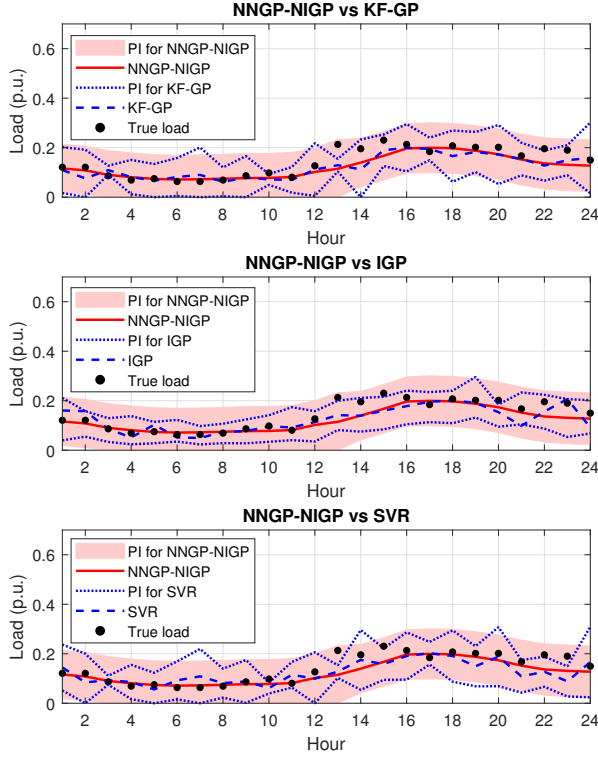


Figure 7: The point and 95% interval predictions obtained by NNGP-NIGP and the benchmarking methods for a customer on an arbitrarily chosen prediction day in the single-phase 8-bus test case.

Table 1

Estimated total Sobol' indices $\hat{S}_{T_j}^2$'s for the inputs $\theta_{1,2}, \theta_{1,3}, \dots, \theta_{1,8}$ in the second stage of NNGP-NIGP. Predictions are performed for customer 1 on 3 days in the single-phase 8-bus test case.

$\hat{S}_{T_j}^2$	θ	Day 1	Day 2	Day 3
$\hat{S}_{T_2}^2$	$\theta_{1,2}$	0.834	0.800	0.897
$\hat{S}_{T_3}^2$	$\theta_{1,3}$	0.015	0.021	0.002
$\hat{S}_{T_4}^2$	$\theta_{1,4}$	2.568×10^{-10}	0.084	1.415×10^{-9}
$\hat{S}_{T_5}^2$	$\theta_{1,5}$	3.435×10^{-11}	1.348×10^{-9}	1.558×10^{-10}
$\hat{S}_{T_6}^2$	$\theta_{1,6}$	2.316×10^{-16}	9.519×10^{-4}	9.984×10^{-5}
$\hat{S}_{T_7}^2$	$\theta_{1,7}$	7.247×10^{-11}	1.460×10^{-4}	0.004
$\hat{S}_{T_8}^2$	$\theta_{1,8}$	5.692×10^{-11}	2.690×10^{-6}	3.468×10^{-10}

1 via (26), then sort the H_ℓ 's in a non-increasing order and only retain the first N^* (recall from Section 3.2) corresponding components in $(\theta_{i,1}, \dots, \theta_{i,i-1}, \theta_{i,i+1}, \dots, \theta_{i,N})^\top$ for input prediction via NNGP. Regarding applying GSA in the second stage to reduce the computational cost of training the NIGP model, one can calculate $\hat{S}_{T_j}^2$'s as detailed in Section 3.3 and select the top ranked components of the input vector for load forecasting via NIGP; the number of inputs re-

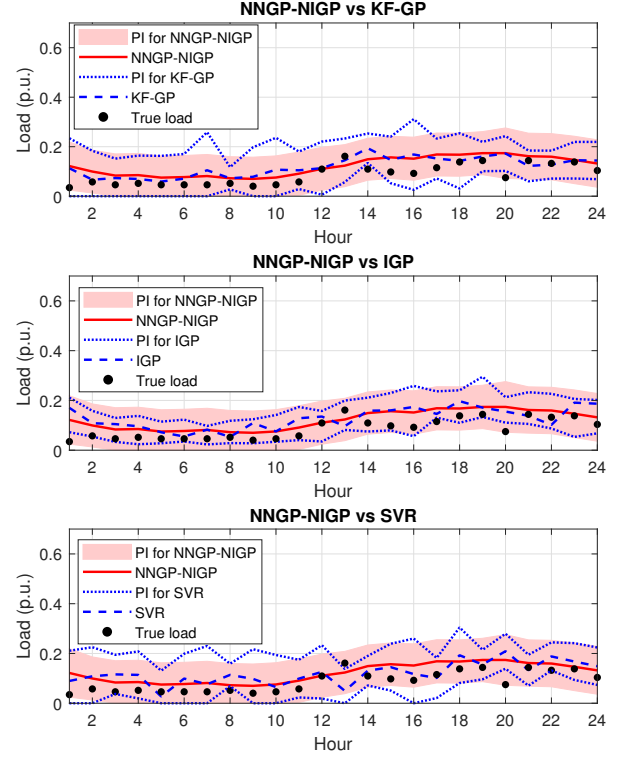


Figure 8: The point and 95% interval predictions obtained by NNGP-NIGP and the benchmarking methods for a customer on an arbitrarily chosen prediction day in the single-phase 24-bus test case.

Table 2

The average computational times (in seconds) of NNGP-NIGP with and without GSA being applied for one day's prediction in all single-phase test cases. The numerical experiments are performed on a laptop with 6th generation Intel® Core™ i7 processor and 8.0GB DDR4 memory.

Scale	NNGP-NIGP w/t GSA	NNGP-NIGP with GSA
8-bus	53.694s	2.469s ~ 4.228s
24-bus	136.736s	3.104s ~ 43.583s
123-bus	241.381s	3.617s ~ 176.736s

tained can be specified based on the user's preference. The aforementioned process can be carried out sequentially for all customers (and for each phase in case of a multi-phase system). For the sake of brevity, the implementation details of the proposed GSA method for other test cases are omitted.

Summary of results. Table 1 shows the estimated total Sobol' indices obtained in the second stage of NNGP-NIGP for load prediction of customer 1 on three different days in the single-phase 8-bus test case. Since the estimated Sobol' index value corresponding to the input $\theta_{1,2}$ is significantly higher than those for the other inputs, $\theta_{1,2}$ is identified as the most important input for load prediction of customer 1. This con-

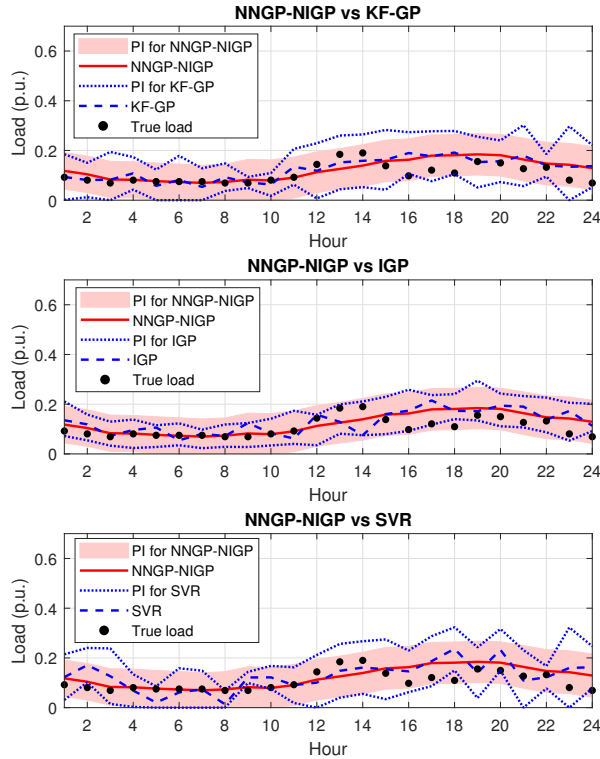


Figure 9: The point and 95% interval predictions obtained by NNGP-NIGP and the benchmarking methods for a customer on an arbitrarily chosen prediction day in the single-phase 123-bus test case.

clusion is consistent with that of IGP given in [15], where a heuristic feature selection method relying on the automatic relevance determination kernel is used. As presenting the GSA results for the other test cases would require considerable space, we mention without showing details that the GSA results are highly consistent with those obtained by IGP in other test cases as well. Table 2 shows the average computational times consumed by NNGP-NIGP with and without GSA being applied in all single-phase test cases. It can be seen that the computational time of NNGP-NIGP is significantly reduced when only those important input features are used for load forecasting.

5. Conclusions

This work represents one of the first efforts to thoroughly address the impact of input errors on both point and interval load predictions for target customers via state-of-the-art machine learning methods. In particular, this work identifies and verifies the opportunity of improving load forecasting performance by incorporating suitable input modeling and uncertainty quantification in a two-stage approach. The first stage delivers point and interval estimates for future input feature values, which are to be used in the second-stage model. The second stage propagates the impact of input er-

rors into the ultimate point and interval load predictions for the target customer. A model-free functional GSA method is proposed to reduce the input-space dimensionality for an enhanced computational efficiency. The numerical experiments demonstrate the superiority of the proposed two-stage approach to three competing methods in terms of point and interval load predictions as well as the significant computational gain achieved by applying the proposed GSA method.

Inaccurate forecasting can result in enormous economic losses to electric power companies, leading to increased operating costs. It has been reported that there is a growth of 10 million operating costs annually associated with 1% raise in forecasting error [41]. This work demonstrates that incorporating appropriate input modeling and uncertainty quantification components into the load forecasting process can significantly enhance the predictive performance. The proposed two-stage approach in conjunction with the GSA method is expected to provide strong support for achieving timely and reliable power systems planning with considerable economic benefits.

Acknowledgment

The work of the first two authors was supported by the National Science Foundation [grant numbers IIS-1849300, CMMI-1846663].

References

- [1] N. Amral, C. S. Ozveren, D. King, Short term load forecasting using multiple linear regression, in: Proceedings of the 42nd International Universities Power Engineering Conference, 2007, pp. 1192–1198.
- [2] T. Hong, M. Gui, M. E. Baran, H. L. Willis, Modeling and forecasting hourly electric load by multiple linear regression with interactions, in: Proceedings of the 2010 IEEE Power and Energy Society General Meeting, 2010, pp. 1–8.
- [3] C.-M. Lee, C.-N. Ko, Short-term load forecasting using lifting scheme and ARIMA models, *Expert Systems with Applications* 38 (2011) 5902–5911.
- [4] D. Alberg, M. Last, Short-term load forecasting in smart meters with sliding window-based ARIMA algorithms, *Vietnam Journal of Computer Science* 5 (2018) 241–249.
- [5] Y. Yang, J. Che, Y. Li, Y. Zhao, S. Zhu, An incremental electric load forecasting model based on support vector regression, *Energy* 113 (2016) 796–808.
- [6] R. K. Jain, K. M. Smith, P. J. Culligan, J. E. Taylor, Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy, *Applied Energy* 123 (2014) 168–178.
- [7] C. Tian, J. Ma, C. Zhang, P. Zhan, A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network, *Energies* 11 (2018) 3493–3505.
- [8] L. Wu, C. Kong, X. Hao, W. Chen, A short-term load forecasting method based on GRU-CNN hybrid neural network model, *Mathematical Problems in Engineering* 2020 (2020) 1–10, Article ID 1428104.
- [9] M. Blum, M. Riedmiller, Electricity demand forecasting using Gaussian processes, in: Proceedings of the 15th Association for the Advancement of Artificial Intelligence Conference on Trading Agent Design and Analysis, 2013, pp. 10–13.
- [10] A. K. Prakash, S. Xu, R. Rajagopal, H. Y. Noh, Robust building energy load forecasting using physically-based kernel models, *Energies* 11 (2018) 1–21, Article ID 862.

- [11] A. Tarsitano, I. L. Amerise, Short-term load forecasting using a two-stage sarimax model, *Energy* 133 (2017) 108–114.
- [12] M. Bozic, M. Stojanovic, Z. Stajic, D. Tasic, A new two-stage approach to short term electrical load forecasting, *Energies* 6 (2013) 2130–2148.
- [13] H. Takeda, Y. Tamura, S. Sato, Using the ensemble Kalman filter for electricity load forecasting and analysis, *Energy* 104 (2016) 184–198.
- [14] S. J. Julier, J. K. Uhlmann, Unscented filtering and nonlinear estimation, in: *Proceedings of the IEEE*, Vol. 92, 2004, pp. 401–422.
- [15] G. Xie, X. Chen, Y. Weng, An integrated Gaussian process modeling framework for residential load prediction, *IEEE Transactions on Power Systems* 33 (2018) 7238–7248.
- [16] Y. Liao, Y. Weng, G. Liu, Z. Zhao, C. Tan, R. Rajagopal, Unbalanced multi-phase distribution grid topology estimation and bus phase identification, *IET Smart Grid* 2 (2019) 557–570.
- [17] R. A. Horn, C. R. Johnson, *Matrix Analysis*, 2nd Edition, Cambridge University Press, New York, NY, 2012.
- [18] A. Monticelli, *State Estimation in Electric Power Systems: A Generalized Approach*, Kluwer Academic Publishers, Norwell, MA, 1999.
- [19] J. Xie, T. Hong, Variable selection methods for probabilistic load forecasting: Empirical evidence from seven states of the united states, *IEEE Transactions on Smart Grid* 9 (2018) 6039–6046.
- [20] A. McHutchon, *Nonlinear modelling and control using Gaussian processes*, Ph.D. thesis, University of Cambridge (2014).
- [21] J. Lee, Y. Bahri, R. Novak, S. S. Schoenholz, J. Pennington, J. Sohl-Dickstein, Deep neural networks as Gaussian processes, in: *Proceedings of the International Conference on Learning Representations*, 2018, pp. 1–17.
- [22] G. Xie, X. Chen, Y. Weng, Input modeling and uncertainty quantification for improving volatile residential load forecasting (2019). URL <https://arxiv.org/abs/1905.06773>
- [23] A. McHutchon, C. E. Rasmussen, Gaussian process training with input noise, in: *Proceedings of Neural Information Processing Systems*, 2011, pp. 1–9.
- [24] C. E. Rasmussen, C. K. I. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, Cambridge, MA, 2006.
- [25] A. McHutchon, NIGP MATLAB package. URL <http://mlg.eng.cam.ac.uk/?portfolio=andrew-mchutchon>.
- [26] A. Shekhovtsov, B. Flach, M. Busta, Feed-forward uncertainty propagation in belief and neural networks, *arXiv:1803.10590v2 [stat.ML]* 1 Nov 2018 (2018) 1–31.
- [27] W. Wright, G. Ramage, D. Cornford, I. Nabney, Neural network modelling with input uncertainty: theory and application, *Journal of Signal Processing Systems* 26 (2000) 169–188.
- [28] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, *Journal of Machine Learning Research* 3 (2003) 1157–1182.
- [29] V. Kumar, S. Minz, Feature selection: a literature review, *Smart Computing Review* 4 (2014) 211–229.
- [30] B. Iooss, P. Lemaître, *A review on global sensitivity analysis methods*, Springer US, Boston, MA, 2015, pp. 101–122.
- [31] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, S. Tarantola, Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index, *Computer Physics Communications* 181 (2010) 259–270.
- [32] A. Marrel, B. Iooss, M. Jullien, B. Laurent, E. Volkova, Global sensitivity analysis for models with spatially dependent outputs, *Environmetrics* 22 (2011) 383–397.
- [33] S. Kucherenko, S. Tarantola, P. Annoni, Estimation of global sensitivity indices for models with dependent variables, *Computer Physics Communications* 183 (2012) 937–946.
- [34] S. Nanty, C. Helbertz, A. Marrelx, N. Perotx, C. Prieur, Sampling, metamodeling, and sensitivity analysis of numerical simulators with functional stochastic inputs, *SIAM/ASA Journal on Uncertainty Quantification* 4 (2016) 636–659.
- [35] M. Jansen, Analysis of variance designs for model output, *Computer Physics Communications* 117 (1999) 35–43.
- [36] Pennsylvania-New Jersey-Maryland system operator. [link]. URL <http://www.pjm.com>
- [37] New York Independent System Operator. [link]. URL <http://www.nyiso.com>
- [38] Renewables.ninja. [link]. URL <https://www.renewables.ninja>
- [39] R. D. Zimmerman, C. E. Murillo-Sanchez, R. J. Thomas, MATPOWER’s extensible optimal power flow architecture, in: *Proceedings of the 2009 IEEE Power and Energy Society General Meeting*, 2009, pp. 1–7.
- [40] R. D. Zimmerman, C. E. Murillo-Sanchez, MATPOWER, a MATLAB power system simulation package (2016). URL <https://matpower.org/docs/MATPOWER-manual-6.0b1.pdf>.
- [41] L. Xiao, J. Wang, X. Yang, L. Xiao, A hybrid model based on data preprocessing for electrical power forecasting, *International Journal of Electrical Power & Energy Systems* 64 (2015) 311–327.
- [42] B. Dhaval, A. Deshpande, Short-term load forecasting with using multiple linear regression, *International Journal of Electrical and Computer* 10 (2020) 3911–3917.
- [43] B. Nepal, M. Yamaha, A. Yokoe, T. Yamaji, Electricity load forecasting using clustering and ARIMA model for energy management in buildings, *Japan Architectural Review* 3 (2020) 62–76.
- [44] Z. Hu, Y. Bao, T. Xiong, Electricity load forecasting using support vector regression with memetic algorithms, *The Scientific World Journal* 2013 (2013) 1–10.
- [45] N. Ghadimi, A. Akbarimajd, H. Shayeghi, O. Abedinia, Two stage forecast engine with feature selection technique and improved meta-heuristic algorithm for electricity load forecasting, *Energy* 161 (2018) 130–142.
- [46] C.-N. Ko, C.-M. Lee, Short-term load forecasting using SVR (support vector regression)-based radial basis function neural network with dual extended Kalman filter, *Energy* 49 (2013) 413–422.

Table 3

Summary of representative one-stage and two-stage load forecasting methods.

Methods		Reference/Year	Highlights	Remark
One-stage	Multiple Linear Regression (MLR)	[1]/2007	MLR model with polynomial terms built on temperature data	MLR models are easy to implement with a high interpretability. However, they must be combined with other sophisticated models such as SVR and NN to perform well.
		[2]/2010	MLR model with a good interpretability to investigate the causality of the consumption of electric energy	
		[42]/2020	MLR model incorporating temperature, due point, load of prior day, hours, and load of prior week as input variables	
	Autoregressive Integrated Moving Average (ARIMA)	[3]/2011	ARIMA model with wavelet multi-resolution analysis	ARIMA models can fit stationary time series well, but do not perform adequately when load series exhibit nonstationarity.
		[4]/2018	Integrated non-seasonal and seasonal sliding window-based ARIMA model using the online information network technique	
		[43]/2020	ARIMA model with k -means clustering for peak load forecasting	
	Support Vector Regression (SVR)	[5]/2016	An incremental learning-based SVR model using batch arriving and large datasets	SVR and NN models are known for their high predictive accuracy when modeling complex nonlinear input-output relationships, but their major drawback is the high computational cost required for model training.
		[6]/2014	SVR model built on sensor data using temperature and solar flux as input variables	
		[44]/2013	SVR model with a firefly-based memetic algorithm for parameter tuning	
	Neural Network (NN)	[7]/2018	Hybrid of convolutional NN for extracting the local trends and long short-term memory NN for learning the relationship between time steps	
		[8]/2020	Hybrid of convolutional NN and gated recurrent unit NN for extracting the feature vector from high-dimensional data and from time sequence data, respectively	
Two-stage	Gaussian process (GP)	[9]/2013	GP model with a task-specific covariance kernel that incorporates seasonal and weather information	GP models provide accurate point predictions and interval predictions that cover the true loads observed with a desirable high probability. But they are computationally costly when tackling large datasets.
		[10]/2018	GP model incorporating physical insights about load data characteristics to improve predictive accuracy	
		[11]/2017	In the first stage, electricity load signal is filtered by some feature selection technique to select appropriate candidates, which are used as the inputs to the forecast engine in the second stage.	Two-stage methods typically produce a higher predictive accuracy due to the first-stage input processing. Existing one-stage and two-stage methods do not address the impact of input errors on load forecasts adequately, however.
		[12]/2013	The first stage performs the next-day average load forecasting, which is used as the input for the next-day hourly load forecasting in the second stage.	
		[45]/2018	Regression parameters and residual series are obtained in the first stage, and a SARIMA model is constructed on the residual process in the second stage.	
		[15]/2018	The first stage forecasts next-hour voltage angle differences via k -means clustering, which are used as the inputs to perform the next-hour load forecasting via standard GP in the second stage.	
		[46]/2013	The first stage employs an SVR model with dual extended Kalman filter to determine the optimal parameters of a radial basis function NN; the second stage adopts the optimal radial basis function NN for short-term load forecasting.	
		[13]/2016	The first stage adopts state-space models with extended Kalman filter to model the load structure, which is used as the input to MLR models to further enhance the forecast accuracy in the second stage.	

A. Literature Review