8-9-2022

Short-term electricity price point and probabilistic forecasts

Chenxu Zhang
czhang84@hawk.iit.edu

Short-term electricity price point and probabilistic forecasts

By

Chenxu Zhang

Approved by:
Yong Fu (Major Professor)
Masoud Karimi
Seungdeog Choi
Bo Tang
Qian Du (Graduate Coordinator)
Jason M. Keith (Dean, Bagley College of Engineering)

A Dissertation
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in Electrical and Computer Engineering
in the Department of Electrical and Computer Engineering

Mississippi State, Mississippi
August 2022

Name: Chenxu Zhang

Date of Degree: August 9, 2022

Institution: Mississippi State University

Major Field: Electrical and Computer Engineering

Major Professor : Yong Fu

Title of Study: Short-term electricity price point and probabilistic forecasts

Pages in Study: 101

Candidate for Degree of Doctor of Philosophy

Accurate short-term electricity price forecasts are essential to all electricity market participants. Generation companies adopt price forecasts to hedge generation shortage risks; load serving entities use price forecasts to purchase energy with low cost; and trading companies utilize price forecasts to arbitrage between markets.

Currently, researches on point forecast mainly focus on exploring periodic patterns of electricity price in time domain. However, frequency domain enables us to identify more information within price data to facilitate forecast. Besides, price spike forecast has not been fully studied in the existing works. Therefore, we propose a short-term electricity price forecast framework that analyzes price data in frequency domain and consider price spike predictions. First, the variational mode decomposition is adopted to decompose price data into multiple band-limited modes. Then, the extended discrete Fourier transform is used to transform the decomposed price mode into frequency domain and perform normal price forecasts. In addition, we utilize the enhanced structure preserving oversampling and synthetic minority oversampling technique to oversample price spike cases to improve price spike forecast accuracy.

In addition to point forecasts, market participants also need probabilistic forecasts to quantify prediction uncertainties. However, there are several shortcomings within current researches. Although wide prediction intervals satisfy reliability requirement, the over-width intervals incur market participants to derive conservative decisions. Besides, although electricity price data follow heteroscedasticity distribution, to reduce computation burden, many researchers assume that price data follow normal distribution. Therefore, to handle the above-mentioned deficiencies, we propose an optimal prediction interval method. 1) By considering both reliability and sharpness, we ensure the prediction interval has a narrow width without sacrificing reliability. 2) To avoid distribution assumptions, we utilize the quantile regression to estimate the bounds of prediction intervals. 3) Exploiting the versatile abilities, the extreme learning machine method is adopted to forecast prediction intervals.

The effectiveness of proposed point and probabilistic forecast methods are justified by using actual price data from various electricity markets. Comparing with the predictions derived from other researches, numerical results show that our methods could provide accurate and stable forecast results under different market situations.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

From a length point of view, price forecasts can be categorized into short-term price forecasts, medium-term price forecasts, and long-term price forecasts. A majority of U.S. electricity markets have a two-settlement markets system that contains a day-ahead (DA) forward market and a real-time (RT) spot market [1]. Price forecasts related to those two markets are regarded as short-term forecasts. Medium-term price forecasts are range from one week to several months [2] and long-term electricity price forecasts focus on predicting electricity prices in the next few years [3]. Although medium-term and long-term price forecasts are vital to energy purchase and policy-making, almost all market participants need to join the day-ahead and the real-time market on a daily basis. Therefore, our research focuses on study short-term electricity price forecasts.

1.1 Point forecast

Short-term (day-ahead and real-time) electricity price forecasts are essential to all electricity market participants. For generation companies, accurate short-term price forecasts are used to derive optimal bidding strategies to hedge risks and reduce losses [4], [5]; for load serving entities, accurate short-term price forecasts can assist them to obtain sufficient energy with low costs [6] [85]; and for trading companies, profitable bidding strategies rely on accurate price forecasts [7].

Researchers have stated that short-term electricity prices are a type of time-series data that possess both periodic and uncertain characteristics [8]. Daily, weekly, seasonally, and other common periodic patterns are used in price forecasts. Reference [9] applies the weighted nearest neighbors' methodology in the day-ahead price forecast, which is a type of similar day forecast method that assumes that if the prices of the previous day i are close to the prices of the current day d, then the prices of day i+1 in the past are similar to the prices of day d+1 in the future. To learn non-linear relationships between predictors and targets that are hard to be captured by a similar day method, a hybrid method is presented in [10] [44] that combines the artificial neural network (ANN) model with a similar day method. Reference [11] uses an autoregressive integrated moving average (ARIMA) method to model linear daily and weekly patterns and implements ANN models to capture nonlinear relationships within short-term electricity prices. Comparing with [11] and [83] that assume error terms with zero mean and constant variance, [4] considers that short-term electricity prices have a seasonality and time-varying nature of volatility. Therefore, the generalized autoregressive conditional heteroskedastic (GARCH) method is adopted to model the time-varying volatility of Spain and California day-ahead electricity prices. By taking advantage of the ARIMA method in modeling stationary linear relationships, the GARCH method in modeling non-constant variance, and the adaptive wavelet neural network (AWNN) method in modeling non-linear relationships, [12] proposes a hybrid model to forecast the day-ahead electricity prices. Comparing with [10], forecast results of the hybrid method in [12] are not only accurate but also stable, which means forecast results of [12] have lower forecast errors and lower error variance. However, training ANN networks is still a time-consuming procedure. To shorten training period and maintain forecast accuracy, [5] utilizes the extreme learning machine (ELM) method to substitute ANN. As input weights of the

ELM method are only generated once instead of being tuned iteratively, compared with the ANN method, the ELM method saves a lot of computation time.

Uncertainty is another characteristic of electricity price data and price spikes are one of the most salient uncertain situations [68]. Normally, price spikes are rarely occurred and have no stable patterns. Besides, the magnitude of spike price is tens or hundreds of times higher than normal electricity prices. Therefore, the difficulties of price spike forecasts are related to price spike occurrence forecast and price spike magnitude prediction. Reference [13] explicitly focuses on forecasting the incidence of price spikes. The probabilistic neural network (PNN) is used to find relationships between the inputs and outputs. Besides, a mutual information (MI) based feature selection method is applied to identify the most useful input data to reduce data redundancy and improve forecast accuracy. Using intra-hour data and applying a lower spike identification threshold, [14] provides a framework to forecast a one-hour ahead spike occurrence. By comparing support vector machine (SVM) and probability classifier, [15] proves that SVM can provide a more reliable spike occurrence prediction. Reference [16] compares a broader range of spike forecast methods than [15] and also concluded that SVM is a better choice to predict spike occurrence. On the other hand, researches also focus on spike magnitude prediction. Reference [17] utilizes a consecutive method in which a naive Bayesian classifier is used to roughly classify spikes into several coarse categories (e.g., prices between [75, 100], [100, 150] … [500, 2000]) and a nearest neighboring hybrid method is applied to calculate the final forecast spike values by averaging k nearest prices in the same category. Researchers also propose a set of price spike forecast methods [18] - [21] that not only predict spike occurrence but also forecast spike values. Reference [18] offers a framework that combines forecast results of the SVM and the Bayesian classifier to improve occurrence forecast accuracy. Besides, spike

magnitudes are estimated by using the nearest neighboring method in [17]. Combined with the spike classification results derived from [13], [19] proposes a hybrid neuro-evolutionary system (HNES) to predict the values of spikes under different spike definitions (e.g., prices over 150 $/MWh or 200 $/MWh). In order to forecast spike values, the HNES framework adopts neural networks (NNs) to estimate spike magnitudes and uses evolutionary algorithms (EA) to tune the parameters of NNs. Reference [20] forecasts the spike occurrence by using a spike classifier and estimates spike values by using a neural networks method. Reference [21] adopts an autoregressive conditional hazard model to forecast spike occurrence in which the ARIMA method is used to estimate spike occurrence by considering the duration of spikes. And, the asymmetric loss scores are used to evaluate the model performance where higher penalties are assigned to spike value forecast errors while normal prices forecast errors have lower penalties.

To some extent, all methods discussed above utilized periodicities resided in price data to perform forecasts. The ARIMA-based methods explicitly express periodicities in formulations. The machine learning based methods use historical price data to train networks that implicitly estimate periodicities. However, both types of methods estimate periodicities through a trial-and-error process which may overlook some important information and result in forecast accuracy degradation. On the other hand, as price spike occurrence is low and historical data is limited, it is hard for those algorithms to give reliable price spike predictions. Therefore, to identify periodicity information of electricity prices in frequency domain and to increase price spike forecast accuracy, we propose a new short-term electricity price forecast framework that analyzes and predicts electricity prices in frequency domain and uses oversampling methods to facilitate price spike forecast. The main contributions are summarized as follows:

1) Decompose the frequency mixed time-series electricity prices into a set of modes by using the variational mode decomposition (VMD) method. To separately analyze periodicity information in electricity price data and reduce random noise, the VMD method is utilized to decompose electricity prices into several modes. The mode is defined as time series data with limited bandwidth in frequency domain and has a center frequency.
2) Use the extended discrete Fourier transform (EDFT) method [22] to forecast normal prices for each mode. There are two reasons to choose the EDFT method. The first one is that the EDFT method achieves a higher resolution in frequency domain than the discrete Fourier transform (DFT) method. This feature helps us to identify the actual frequencies within electricity price data. The second one is that the EDFT method does not have limitations on the forecast data, while the DFT method assumes that the data outside the observation period repeat the data within the observation period.
3) Synthesize price spike cases to improve price spike forecast accuracy. As limited real cases hinder spike forecast accuracy, we use the following methods to increase the number of spike cases. Spike cases contain two parts. The predictor variables are the inputs of spike cases and the target values are the outputs. We oversample predictors by using the enhanced structure preserving oversampling (ESPO) method to ensure that the synthesized cases have similar temporal structures as the real spike cases. The target values are oversampled by utilizing the synthetic minority oversampling technique (SMOTE) for regression.

1.2 Probabilistic forecast

Currently, there are two major electricity price forecast types. One is the point forecast, and the other is the probabilistic forecast [50]. The characteristic of point forecast is that algorithms only give one prediction at each time point in the future. Point forecast has several accretive prosperities. The most salient one is that the point forecast results are easy to understand. Besides, under ideal conditions, point forecasts are able to provide an accurate and parsimonious relationship between predictors and target variables [51]. However, there is an inherent limitation within point forecasts [62]. As the relationships between predictors and target variables are not fixed, so the same predictors could result in different target values [63]-[64], [66]. Therefore, in order to acquire a deterministic prediction, point forecast algorithms normally look for the conditional mean function to link between predictors and target variables [71] [74]. Because of this limitation, point forecast algorithms cannot provide the distribution of target values related to predictors, and price uncertainty information also cannot be provided [82]. Besides, as a majority of day-ahead and real-time electricity prices belong to the low-price zone and only a few prices reside in the high-price zone, the distribution of electricity prices is heavy-tailed. Therefore, point forecasts could be inaccurate as conditional means might be affected by a handful of price spikes.

On the other hand, probabilistic forecasts are able to provide predictions of electricity price distribution that resolve the difficulties reside in point forecasts [50] and [69]. Normally, the probabilistic forecasts can be categorized into two groups that one is the probability density forecast and the other is the quantile or prediction interval (PI) forecast [8] [81][84]. Probability density forecast aims to provide all distributional information of electricity prices to incorporate uncertainties in the future. As total distribution is hard to obtain, usually, probability density

forecast assumes the predictive errors follow a normal distribution or logistic distribution [8] [57] [61] [72] [76] [78]. The assumption is reasonable when data are normally distributed. However, since electricity prices are a set of highly skewed data, the normal distribution assumption does not fit electricity price data well and leads to poor distribution predictions [52]. Different from the probabilistic density forecast, the prediction interval forecast aims to predict an interval that the actual data will fall into with a predefined probability [7]. Researches have shown that prediction interval is able to circumvent predefined data distribution assumption. Methods include using quantile regression to estimate the upper and lower bound of the interval or using the kernel density estimation method to provide price distribution [53]. Therefore, considering the heavy-tailed feature of electricity price distribution, we decide to use the prediction interval to quantify the uncertainty of electricity prices.

Generally, we use predictors to forecast target values. The relationships between predictors and targets can be classified into linear and nonlinear categories. The autoregressive integrated moving average (ARIMA) based methods are able to model linear relationships and are widely used in the time series analysis and electricity price point forecasts. Zhou et al. [54] extended the ARIMA method to enable the proposed framework can provide probability forecasts. They first use the ARIMA method to provide electricity price point forecasts. Then, forecast errors are receptively updated. The forecast errors are derived by using the actual electricity prices to minus the point and the error predictions. The error update stops when the confidential interval of error distribution meets the pre-defined criteria. The benefit of the proposed method is easy to implement. However, in order to facilitate distribution calculation, this method has some limitations. To derive confidential interval, the proposed method assumes that the distribution of error follows a Gaussian or uniform distribution. The price distribution

assumption might not fit the deregulated markets electricity prices as price distribution is highly skewed. To avoid pre-defining price distribution, Weron et al. [55] utilizes a semi-parameter time series model to estimate prediction intervals. In their research, although the point forecasts are still derived from ARIMA-based methods, the parameters of ARIMA models are derived by maximizing the likelihood of error distribution, and the electricity price forecast error density function is calculated by using kernel density methods such as Hsieh–Manski Estimator or Kreiss' Estimator. Compared with [54], they resolve the distribution assumption problem. However, the ARIMA-based methods are hard to model nonlinear relationships between predictors and target values.

In order to capture nonlinear relationships between predictors and target values [70], several probabilistic forecast methods adopt machine learning in their studies. Shrivastava et al. [38] apply the support vector machines (SVM) to learn the nonlinear relationships from the training data and applied the trained SVMs to forecast prediction intervals. Besides, the particle swarm optimization method (PSO) is used to tune the hyper-parameters of the SVMs. In order to derive the bounds of prediction intervals, they use two SVM models to forecast the upper and lower bound. Compared with [38], Khosravi et al. [43] use a neural network to perform electricity price point forecasts and adopt a bootstrap method to forecast prediction intervals. Instead of directly constructing the upper and lower bound of PI, Khosravi et al. [43] assume forecasts follow a Gaussian distribution. The variance of the distribution comprises two parts the one is the model variance and the other one is the noise variance. The model variance is calculated by averaging the prediction variance of all bootstrap models, and a separate neural network [56] predicts the noise variance. As the model variance and the noise variance are statistically independent, the total variance equals the summation of those two variances.

Because of the Gaussian distribution assumption, the upper and lower bound of PI are calculated by using the point forecasts adding or subtracting the critical value of the normal distribution multiplying the square root of the total variance, respectively. In [7], the authors propose a method to forecast prediction intervals that combine ideas from [43] and [38]. To provide electricity price point forecasts, they adopt a SVM method to handle the nonlinear relationship between predictors and target values that is the same as [38]. To forecast prediction interval, they also assume the total variance is normally distributed and they calculate the upper and lower bound of PI using the same method as [43]. The only difference between [43] and [7] is that Zhao et al. directly provide the total variance forecasts but Khosravi et al. separately estimate the model variance and the noise variance.

Recently, a single hidden layer feed-forward neural network (SLFN) called extreme learning machine (ELM) draws a lot of attention [65]. The ELM is able to model nonlinear relationships as traditional neural networks but the training speed is much faster. In [5], the authors select to use the ELM to provide point forecasts. They calculate prediction intervals by using a bootstrap method. However, the bootstrap method used in [5] is different from the bootstrap method used in [43]. Chen et al. [5] assume that if the bootstrap replications are large enough, the quantiles derived from ELM can form prediction intervals non-parametrically. Therefore, the ELM point forecasts arrange in ascending order, and the upper and lower bound of prediction intervals are the quantiles of the ELM point forecasts. By taking advantage of the fast-training characteristic of the ELM method, [57] also adopt this method to provide prediction interval forecasts. The forecast structure of [57] is the same as [43] that use a bootstrap method to estimate the model variance and adopt a separate neural network to forecast the noise variance. Also, the upper and lower bound of PI are calculated using the same method

as [43]. The only difference between [57] and [43] is that [57] uses the fast-training extreme learning machine to substitute the neural network. Different from methods that directly forecast the bounds of prediction interval or use bootstrap methods to estimate prediction intervals, [58] proposes a method to derive an optimal prediction interval considering both reliability that is used to evaluate PI forecast accuracy and sharpness that is to measure the width of PI prediction [67]. The upper and lower bound of prediction intervals are also forecasted by using the ELM method. To derive optimal intervals, a heuristic method called non-dominated sorting genetic algorithm II (NSGA-II) is adopted.

Except [58], the above prediction interval methods rarely discuss the quality of prediction intervals. Wide intervals imply that the price forecast uncertainty level is high. For market participants, they should make a bid or offer decision with more caution. Therefore, we expect the width of prediction intervals as narrow as possible so that decisions can be made with confidence. Except for narrow width, we also expect that prediction intervals could reach the pre-defined nominal confidence level. Thus, optimal prediction intervals should not only have a narrow width but also meet the pre-defined nominal confidence level criterion. Besides, the speed of forecast algorithms should also be taken into consideration. However, no matter using the PSO to tune hyper-parameters of the SVM or using bootstrap methods to estimate predictions, or applying the heuristic algorithm NSGA-II to perform optimization, they are all time-consuming.

Therefore, there are several major objectives we want to achieve. The first one is to derive a prediction interval forecast method that considers both reliability requirements to ensure forecast accuracy and sharpness requirement to reduce prediction interval width. The second objective is that our forecast method could capture the nonlinear relationship between predictors

and target values. And the third one is that the forecast method should not rely on pre-defined price data distribution assumptions. In order to meet all requirements, we propose an electricity price prediction interval forecast method. The main technical contributions are summarized:

1) We propose an optimal electricity price probabilistic forecast method to estimate prediction interval while considering reliability and sharpness.
2) We adopt the fast-training extreme learning machine (ELM) method to learn the nonlinear relationship between predictors and target values.
3) We utilize the quantile regression method to estimate the upper and lower bound of prediction intervals so that price distribution assumption can be avoided.

1.3 Proposal organization

The whole proposal contains five chapters. In Chapter I, we introduce two types of electricity price forecast that one is the point forecast, and the other is the probabilistic forecast. After reviewing previous researches, major motivations and core contributions of our proposed methods are listed.

In Chapter II, a point forecast framework is presented. The proposed method flowchart is shown in section 2.1. In section 2.2, key methods and algorithms are introduced. Following each algorithm, we use a simple example to demonstrate its function in our proposed method. At the end of this chapter, we provide a summary.

Chapter III is used to evaluate the effectiveness of the proposed price forecast framework. In this chapter, we introduce real market data that are used in our test in section 3.1. In section 3.2, we list some metrics to measure the performance of our forecast results. The normal and price spike prediction results are shown in sections 3.3 and 3.4, respectively. In section 3.5, we

use data from different seasons and from different markets to further justify the performance of our method. We draw conclusions in section 3.6.

In Chapter IV, an optimal prediction interval forecast method is proposed. In section 4.1, we list the metrics that are used to measure the performance of prediction interval forecasts. The formulations of the proposed model and the solving method are demonstrated in section 4.2. Also, the flowchart and detailed calculation procedures are also shown in section 4.2 to facilitate understanding.

In Chapter V, based on preliminary studies and currently obtained numerical results of our researches, the initial conclusions on the point and probabilistic forecast are drawn. Finally, future works are listed at the end of this chapter.

CHAPTER II

PROPOSED POINT FORECAST METHOD

In this section, a forecast framework is presented for the short-term electricity price forecast. The section 2.1 presents the proposed short-term electricity price forecast flowchart. The key algorithms and methods are demonstrated in section 2.2. Following the introduction of each algorithm, a simple example is utilized to show the usage of the algorithm in the forecast framework. At end of chapter II, a brief summary related to the proposed forecast framework is shown in section 2.3.

2.1 Framework of the proposed short-term price forecast

The proposed short-term electricity price forecast framework is shown in Figure 2.1. Its major steps are described as follows (Note that steps 2-4 are for the normal price forecast and steps 5&6 are for the price spike forecast. They can be implemented in parallel):

Step 1: Data Preprocessing: Electricity price data derived from public resources are imported to the proposed price forecast program. As raw data may have some deficiencies (e.g., data loss and data duplication), we need to identify and modify the imperfect data. For example, the missing price data could be substituted by an average price that is derived from price values before and after the missing one; and the duplicated price is eliminated to ensure every time spot has only one price data.

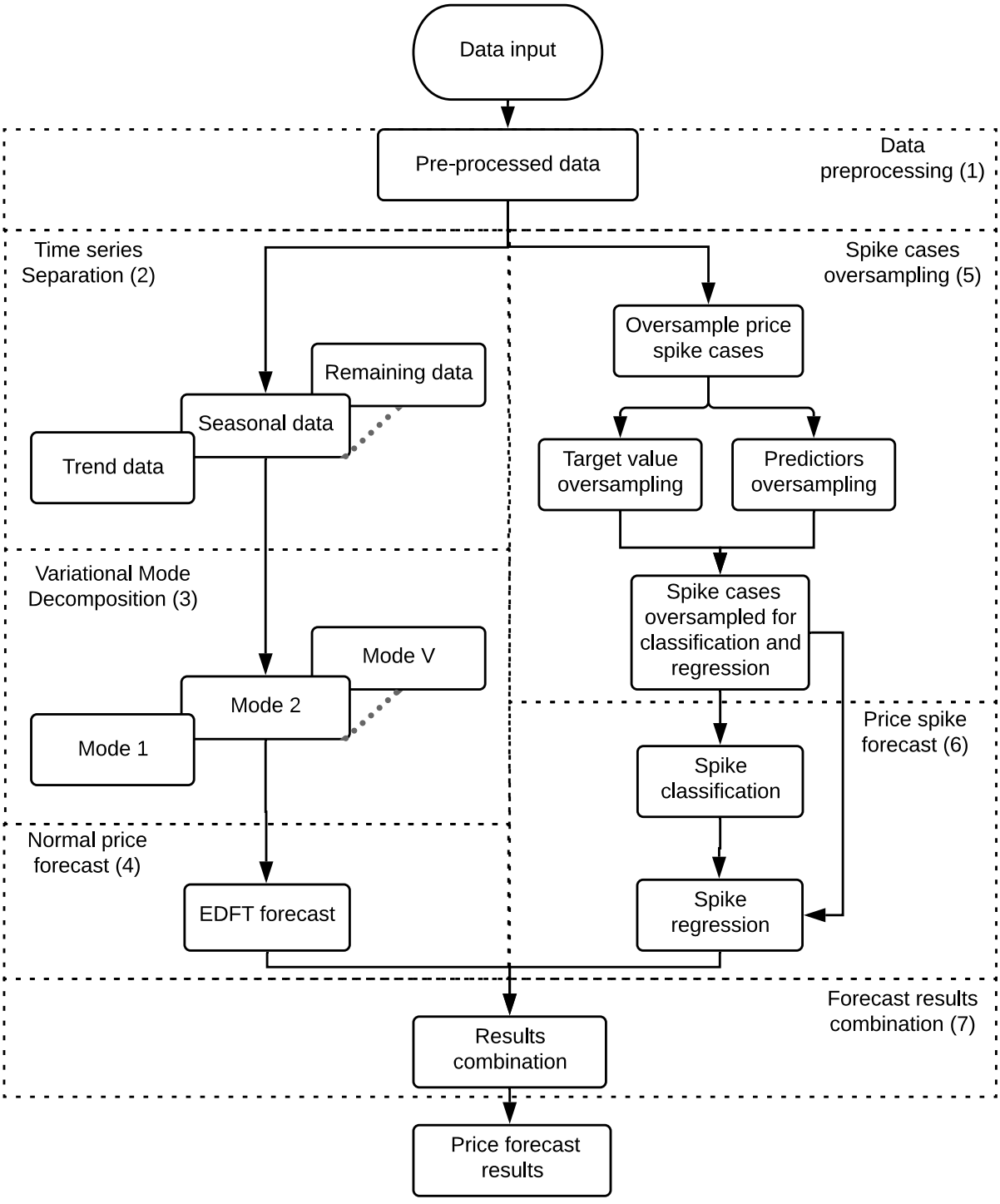


Figure 2.1 Framework of the proposed short-term electricity price forecast

Step 2: Time-series Separation. Electricity prices can be viewed as a set of time-series data that consists of trend and seasonal components [23]. To improve forecast accuracy and facilitate data analysis, we perform a time-series decomposition to separate the trend and seasonal components from the input price data. The trend component, which shows a long-term increase/decrease tendency, is calculated by using the moving average method; the seasonal component, which demonstrates a repeating short-term cycle in the series, is captured by averaging the detrended time-series data for a certain seasonality period; and the remaining data are obtained from the input price data after extracting trend and seasonal components.

Step 3: Variational Mode Decomposition. The price data components (e.g., trend, seasonal, and remaining) discovered from Step 2 are still mixed with frequency components. Although the periodicities of the decomposed time-series price data can be directly utilized to provide forecasts with moderate accuracy, to further improve prediction precision, the frequencies hidden in the decomposed time-series data should be identified. In this step, the VMD method is adopted to transform each price data component into multiple price modes that will facilitate the following normal price forecast module.

Step 4: Normal Price Forecast. The normal electricity price forecast is performed by using the EDFT method as the EDFT method enables us to find the frequencies within electricity prices and utilizes the identified frequencies to provide electricity price predictions. As the time-series price data decomposition and the VMD decomposition are additive models, the outputs of this step are calculated by summing up the EDFT forecast results of all modes and components.

Step 5: Spike Cases Oversampling. As the number of electricity price spikes is much less than normal prices and so the imbalanced dataset will influence the accuracy of price spike forecast, we will synthesize price spike cases. In this step, we first define price spikes for each

hour as electricity prices at different hours have different patterns. Then, we form price spike cases that have two parts. One part is for the target values and the other part is for the predictors. The target values are the price spikes that need to be forecasted and the predictors are the variables used to predict the target values. Then, we oversample the predictors and targets of the price spike cases for the following price spike classification and regression forecast.

Step 6: Price Spike Forecast. A classifier that is tuned to have an acceptable classification accuracy, such as the well-known support vector machine (SVM) [15], [16], is used to identify whether the predicted electricity price is a price spike or not. In order to improve the accuracy of the price spike classification, the classifier is trained by using the oversampled dataset. Then, a regression algorithm (e.g., SVM trained by the oversampled dataset) is used for the identified price spike to predict its value.

Step 7: Forecast Results Combination. This step combines the normal price and price spike prediction results. If a forecasted price is identified by the price spike classification as a spike, we get the price value from the price spike forecast in Step 6. Otherwise, the price value from the normal price forecast in Step 4 is applied. Therefore, the final output of the short-term electricity price forecast framework is the combined forecast results.

2.2 Point forecast key modules

In this section, three key methodologies in the proposed price forecast framework, including VMD, EDFT, and spike oversampling methods, are discussed in detail.

2.2.1 Variational mode decomposition module

As illustrated in Figure 2.1, the VMD [24] method is utilized to decompose the frequency mixed price component data $f(t)$ into several modes where each mode is band-limited data with

a center frequency. The input of the VMD module is the trend, seasonal, or remaining data and its time-series output modes will be used to facilitate normal price forecasts.

Firstly, we transfer the price mode data into its analytical form as this form has three features that benefit frequency analysis. The first feature is that the analytical form of data has a positive unilateral spectrum in frequency domain. If the Fourier transform is directly applied to the price data, we get a spectrum with both positive and negative frequency components. However, by converting price data to its analytical form, the spectrum derived from the Fourier transform only has a positive frequency part. The second feature is that when demodulating price data within frequency domain by shifting frequency spectrum, we only need to consider the shifting effect on a one-sided spectrum. The last feature is that it is easy to restore the original data by only retrieving the real part from the analytical form of data.

By using (2.1), the data $u_v(t)$ of price mode v is able to be transformed into its analytical form $u_{A,v}(t)$,

$$u_{A,v}(t) = u_v(t) + jHu_v(t) = \left(\delta(t) + \frac{j}{\pi t}\right) * u_v(t) \tag{2.1}$$

where H is the Hilbert transform operator (the impulse response is $h(t) = 1/\pi t$ in time domain and $h(\omega) = j\omega/|\omega|$ in frequency domain); $\delta(t)$ is the Dirac delta function; and $*$ is a convolution operator.

After analytical transformation, in order to derive a band limited price mode $u_v(t)$ in frequency domain, we can recenter the spectrum of $u_v(t)$ and apply a low pass filter. The amount of shift for every mode v is determined by each mode's center frequency ω_v. The formulation of recenter process is shown in (2.2).

$$u_{A,v}(t)e^{-j\omega_v t} = \left[\left(\delta(t) + \frac{j}{\pi t}\right) * u_v(t)\right] e^{-j\omega_v t} \tag{2.2}$$

Note that after the Fourier transform, the $e^{-j\omega_v t}$ term in time domain is converted to a Dirac delta $\delta(\omega + \omega_v)$ in frequency domain. Therefore, in (2.2), multiplying the analytical form of mode $u_{A,v}(t)$ by $e^{-j\omega_v t}$ in time domain equals to convoluting $u_{A,v}(\omega)$ with $\delta(\omega + \omega_v)$ in frequency domain. The effect of this convolution is to shift the spectrum $u_{A,v}(\omega)$ to the left by ω_v amount.

The bandwidth of the price mode $u_v(t)$ can be determined by optimizing the Gaussian smoothness of the demodulated spectrum in frequency domain which equals to optimize the squared L^2 norm of the demodulated data's gradient in time domain. Thus, the objective is formed in (2.3). The constraint (2.4) is used to ensure the price component data $f(t)$ can be recovered from the decomposed price modes $u_v(t)$.

$$min\left\{\sum_v \left\|\partial_t\left[\left(\delta(t)+\frac{j}{\pi t}\right)*u_v(t)\right]e^{-j\omega_v t}\right\|_2^2\right\} \tag{2.3}$$

$$s.t.\sum_v u_v(t) = f(t) \tag{2.4}$$

In order to relieve the constraint (2.4), the augmented Lagrangian method is used to convert the above constrained optimization problem to an unconstrained form (2.5) by moving the constraint (2.4) into the objective (2.3)

$$\begin{aligned}\mathcal{L}(u_v, \omega_v, \lambda) = \alpha\sum_v \left\|\partial_t\left[\left(\delta(t)+\frac{j}{\pi t}\right)*u_v(t)\right]e^{-j\omega_v t}\right\|_2^2 \\ + \left\|f(t)-\sum_v u_v(t)\right\|_2^2 + \lambda\left[f(t)-\sum_v u_v(t)\right]\end{aligned} \tag{2.5}$$

, where α is the balancing parameter of the data-fidelity constraint and λ is Lagrangian multiplier in time domain.

Since it is hard to solve the above optimization problem with exponential term multiplication, convolution, and derivative operation in time domain, we transfer (2.5) from time domain into frequency domain by using the Fourier transform.

$$min\ \alpha \sum_{v} j\omega \left\| \left[\left(1 + sgn(\omega + \omega_v)\right) u_v(\omega + \omega_v) \right] \right\|_2^2 + \left\| f(\omega) - \sum_{v} u_v(\omega) + \frac{\lambda(\omega)}{2} \right\|_2^2 \tag{2.6}$$

where the exponential term multiplication becomes the frequency spectrum shift operation; the convolution becomes multiplication; the derivative operation becomes multiplying the objective function with $j\omega$; and the sgn means the sign function.

As each mode in (2.6) can be viewed as an independent frequency spectrum $u_v(\omega)$ with a center frequency ω_v in frequency domain, by using the alternate direction method of multipliers (ADMM) method, the spectrum of mode $u_v(\omega)$ is calculated by iteratively optimizing (2.7).

$$u_v^{r+1}(\omega) = \underset{u_v}{\operatorname{argmin}} \{ \alpha \left\| j\omega \left[\left(1 + sgn(\omega + \omega_v^r)\right) u_v^r(\omega + \omega_v^r) \right] \right\|_2^2 + \left\| f(\omega) - \sum_{l=1}^{v-1} u_l^{r+1}(\omega) - \sum_{l=v}^{V} u_l^r(\omega) + \frac{\lambda^r(\omega)}{2} \right\|_2^2 \} \tag{2.7}$$

where r represents iteration number; l is mode index; and V is the total number of modes. As the price mode data are a set of real numbers, by applying the Hermitian symmetry, the equation (2.7) can be simplified to (2.8) where ω is used to substitute $\omega - \omega_v$ in the first term,

$$u_v^{r+1}(\omega) = \underset{u_v}{\operatorname{argmin}} \{ \int_0^{\infty} 4\alpha(\omega - \omega_v^r)^2 |u_v^r(\omega)|^2 + 2 \left| f(\omega) - \sum_{l=1}^{v-1} u_l^{r+1}(\omega) - \sum_{l=v}^{V} u_l^r(\omega) + \frac{\lambda^r(\omega)}{2} \right|^2 d\omega \} \tag{2.8}$$

The optimal solution of (2.8) is calculated by letting its derivative equal to zero. Thus, the $u_v^{r+1}(\omega)$ can be expressed as,

$$u_v^{r+1}(\omega) = \frac{f(\omega) - \sum_{l=1}^{v-1} u_l^{r+1}(\omega) - \sum_{l=v+1}^{V} u_l^r\,(\omega) + \frac{\lambda^r(\omega)}{2}}{1 + 2\alpha(\omega - \omega_v^r)^2} \tag{2.9}$$

By applying (2.9), the price mode $u_v(\omega)$ is demodulated from the input by using a low-pass narrow-band filter around $\omega = \omega_v$ which is able to remove high frequency noise from the data.

As the center frequency ω_v does not appear in the second term of (2.6), we ignore this term from the center frequency calculation. So, the center frequency ω_v of $u_v(\omega)$ is the optimal solution of (2.10).

$$\omega_v^{r+1} = \underset{\omega_v}{\operatorname{argmin}} \left\{ j\omega \left\| \left[\left(1 + sgn(\omega + \omega_v^r)\right) u_v^{r+1}(\omega + \omega_v^r) \right] \right\|_2^2 \right\} \tag{2.10}$$

Similar to (2.8), the equation (2.10) can be simplified into (2.11)

$$\omega_v^{r+1} = \underset{\omega_v}{\operatorname{argmin}} \left\{ \int_0^{\infty} (\omega - \omega_v^r)^2 |u_v^{r+1}(\omega)|^2 \, d\omega \right\} \tag{2.11}$$

The solution of (2.11) is

$$\omega_v^{r+1} = \frac{\int_0^{\infty} \omega |u_v^{r+1}(\omega)|^2 \, d\omega}{\int_0^{\infty} |u_v^{r+1}(\omega)|^2 \, d\omega} \tag{2.12}$$

Note that Lagrangian multipliers $\lambda(\omega)$ in frequency domain is updated by using (2.13), where τ is the penalty parameter.

$$\lambda^{r+1}(\omega) = \lambda^r(\omega) + \tau(f(\omega) - \sum_v u_v^{r+1}\,(\omega)) \tag{2.13}$$

The above iterative calculations (2.9), (2.12) and (2.13) will run until the convergence criterion (2.14) is met.

$$\frac{\sum_v \|u_v^{r+1}(\omega) - u_v^r(\omega)\|_2^2}{\|u_v^r(\omega)\|_2^2} \leq \varepsilon \tag{2.14}$$

As the output of the VMD module is a set of modes in time domain, by using real part of the inverse Fourier transform in (2.15), we can transfer $u_v(\omega)$ from frequency domain back to time domain, $u_v(t)$.

$$u_v(t) = Real(ift(u_v(\omega))) \tag{2.15}$$

where, the $Real$ is an operator that only takes the real part of the inverse Fourier transform ift.

Example 1: The following example illustrates the effectiveness of the VMD method to separate a set of modes from a frequency mixed time series data. The function of $\cos(2\pi 2t) + 0.5\cos(2\pi 6t) + noise$ is used to generate the time series data, that is composed of two cosine waves with center frequency 2 Hz and 6 Hz, and random noise. The plot of this time series data is shown in Figure 2.2.

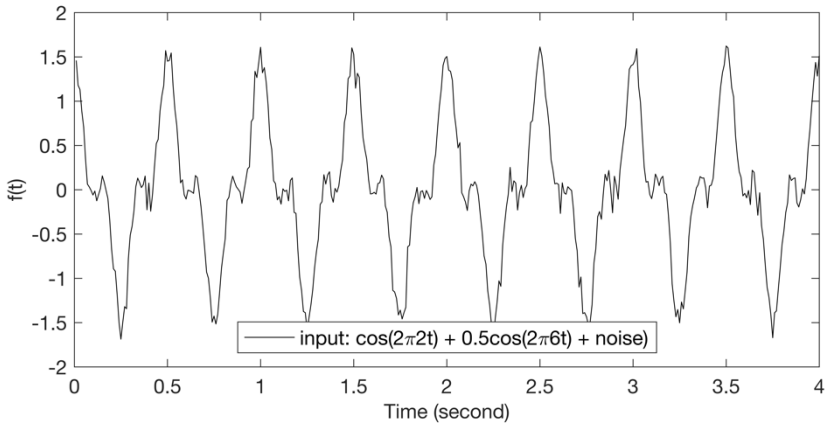


Figure 2.2 Input signal of the VMD method example

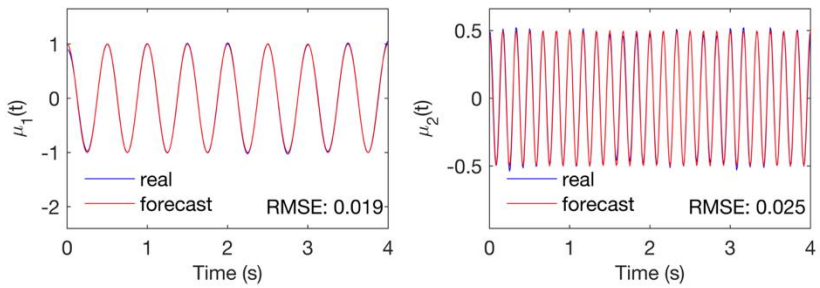


Figure 2.3 Decomposed signals of the VMD method example

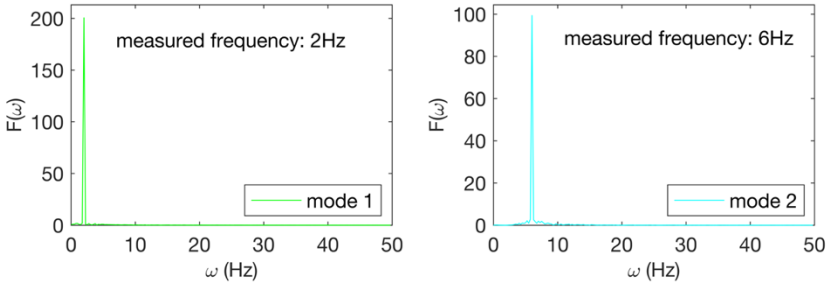


Figure 2.4 Spectrums of the VMD decomposed signals

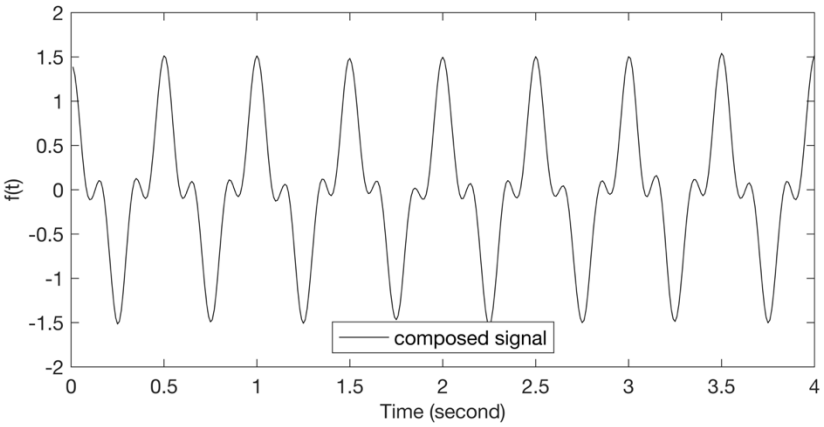


Figure 2.5 Composed signal

After applying the proposed VMD method, the time series input is decomposed into two modes shown in Figure 2.3. To identify whether the decomposed modes are the same as the cosine waves in the input, we use the root-mean-square error (RMSE) metric to calculate the

difference between them. The RMSE values are 0.019 and 0.025 for the $2\ Hz$ and $6\ Hz$ cosine waves, respectively. Besides, to view the decomposed modes from frequency perspective, Figure 2.4 shows the spectrum of each mode. On the left-hand side of Figure 2.4, there is a spike at the frequency $2\ Hz$ that means that the center frequency of this mode is $2\ Hz$. On the right-hand side, the center frequency is $6\ Hz$. The center frequencies of these two modes match the frequencies mixed in the input data. In the end, Figure 2.5 shows the time series data recovered from the decomposed modes. Compared the input plot Figure 2.2 with the recovered plot Figure 2.5, we can tell the recovered time series data is almost identical to the input without noise. Therefore, the VMD method can be utilized to further identify frequencies within the price data components provided by Step 2 in Figure 2.1 and remove noises from the data, which will benefit the forecast.

2.2.2 Extended discrete Fourier transform module

In the proposed price forecast framework, the EDFT method is selected to predict normal prices rather than the DFT method. The first reason is that the frequency resolution of the DFT method is limited by the length of input data. If frequency resolution is not fine enough, a spectrum leakage situation may happen and the true periodicity within price data may not be identified. The second one is that the DFT method simply assumes that data outside the observation period is the same as the data within the period. Although this assumption can ensure the spectrum derived from the DFT method is in a discrete form which can be handled easily by modern computing derives, it may not be true. Therefore, we utilize the EDFT method to provide high frequency resolution around periodicities within the price data without imposing limitations on data outside the observation period. According to Figure 2.1, the input of the EDFT module is

the time series price mode derived from the VMD module and its output is the forecast for each mode.

As the continuous frequency spectrum derived from the Fourier transform provides the finest frequency resolution, we regard the continuous frequency spectrum as the target and use a new basis $\alpha(\omega, t_k)$ to convert data from time domain into frequency domain. To simplify expressions in this section, the symbol $u(t)$ is utilized to substitute the price mode data $u_v(t)$ from (2.15). Since the price mode data $u(t)$ are sampled uniformly, the basis $\alpha(\omega, t_k)$ can be written as $\alpha(\omega, kT)$. So, the EDFT can be written as (2.16)

$$F_\alpha(\omega) = \sum_{k=0}^{K-1} u(kT)\, \alpha(\omega, kT) \tag{2.16}$$

where $u(kT)$ is the mode derived from the VMD module uniformly and $k = 0,1,2 \ldots, K-1$; K is the number of sampled points within the observation period; T is the sampling interval; and ω is the radian frequency.

In order to ensure that the EDFT (2.16) and the Fourier transform (2.17) are as close as possible, we formulate the objective (2.18) where Δ is the squares error expression between (2.17) and (2.16).

$$F(\omega) = \int_{-\infty}^{+\infty} u(t) e^{-j\omega t}\, dt \tag{2.17}$$

$$\Delta = |F(\omega) - F_\alpha(\omega)|^2 \tag{2.18}$$

However, the Fourier transform (2.17) cannot be used directly on the input $u(t)$ as there are no infinite long prices period. To circumvent this obstacle, the input prices with ω_0 frequency can be expressed in the form of (2.19) where $-\infty < t < \infty$,

$$u(\omega_0, t) = S(\omega_0) e^{j\omega_0 t} \tag{2.19}$$

By substituting (2.19) into (2.17), the Fourier transform $F(\omega)$ at the frequency ω_0 can be expressed by the Dirac delta function in (2.20),

$$F(\omega_0) = \int_{-\infty}^{+\infty} u(\omega_0, t) e^{-j\omega t}\, dt = 2\pi S(\omega_0)\delta(\omega - \omega_0) \tag{2.20}$$

where ω_0 is the cyclic frequency within the range $-\Omega \leq \omega_0 \leq \Omega$; Ω is the upper frequency of the prices $f(\omega_0, t)$; and $S(\omega_0)$ is the complex amplitude of $f(\omega_0, t)$ in frequency domain. By using the $F(\omega)$ in (2.20) and the $F_\alpha(\omega)$ in (2.16) where the $u(kT)$ is substituted by using (2.19), the difference between the Fourier transform and the EDFT is shown in (2.21).

$$D = 2\pi S(\omega_0)\delta(\omega - \omega_0) - \sum_{k=0}^{K-1} S(\omega_0) e^{j\omega_0 kT}\, \alpha(\omega, kT) \tag{2.21}$$

So, the objective function (2.18) can be rewritten as (2.22).

$$\Delta = \int_{-\Omega}^{\Omega} |D|^2 d\omega_0 \tag{2.22}$$

To specifically calculate the EDFT basis $\alpha(\omega, kT)$ so as to minimize square error Δ, we can solve (2.22) by letting the derivative of this function with respect to $\alpha(\omega, lT)$ equals to zero, $\partial\Delta/\partial\alpha(\omega, lT) = 0$, $l = 0,1, \ldots K-1$. The solution is shown in (2.23).

$$\int_{-\Omega}^{\Omega} \left[2\pi S(\omega_0)\delta(\omega - \omega_0) - \sum_{k=0}^{K-1} S(\omega_0)\, e^{j\omega_0 kT} \alpha(\omega, kT)\right] S(\omega_0)^* e^{-j\omega_0 lT} d\omega_0 = 0 \tag{2.23}$$

As Dirac delta functions are able to select frequencies, the equation (2.23) can be simplified into (2.24) where the first term of (2.23) is at the right side of (2.24).

$$\sum_{k=0}^{K-1} \left(\frac{1}{2\pi} \int_{-\Omega}^{\Omega} |S(\omega_0)|^2\, e^{j\omega_0 (k-l)T} d\omega_0\right) \alpha(\omega, kT) = |S(\omega)|^2 e^{-j\omega lT} \tag{2.24}$$

The solutions of (2.24) that is the EDFT basis $\alpha(\omega, kT)$ can be expressed (2.25).

$$\boldsymbol{A}_{\omega} = |S(\omega)|^2 \boldsymbol{R}^{-1} \boldsymbol{E}_{\omega} \tag{2.25}$$

where $\boldsymbol{E}_{\omega}$: $e^{-j\omega lT}$, and the components of $\boldsymbol{R}(K \times K)$**,** are expressed as (2.26).

$$r_{l,k} = \frac{1}{2\pi} \int_{-\Omega}^{\Omega} |S(\omega_0)|^2 \, e^{j\omega_0 (k-l)T} d\omega_0 \tag{2.26}$$

Then, we can utilize (2.27) to transform the input price mode in time domain into frequency domain by using the new derived basis.

$$F_{\alpha}(\omega) = \boldsymbol{u}\boldsymbol{A}_{\omega} = |S(\omega)|^2 \boldsymbol{u}\boldsymbol{R}^{-1}\boldsymbol{E}_{\omega}, \qquad -\Omega \le \omega \le \Omega \tag{2.27}$$

To facilitate the above calculation, continuous function integration can be substituted by summation. For instance, the equation (2.26) can be estimated by (2.28) if the number of sampled points N within the integral interval is large enough.

$$r_{l,k} \approx \frac{\Omega}{\pi N} \sum_{n=0}^{N-1} |S(\omega_n)|^2 e^{j\omega_n (k-l)T} \tag{2.28}$$

where - $\Omega \le \omega_n \le \Omega$ $n = 0,1, \ldots N-1$. To derive (2.28) from (2.26), we can regard the integration as to calculate the area under a curve. The inner part of integration is the height and $d\omega_0$ is the width. In order to substitute integration with summation, we have $2\Omega/N$ points between frequency interval $[-\Omega, \Omega]$.

By viewing (2.27), the last obstacle that hinders the EDFT calculation is that the true amplitude spectrum $S(\omega)$ of the input price mode is unknown. As suggested by [22], the amplitude spectrum can be obtained by using an iterative process (2.29) - (2.32). The equation (2.29) is the matrix form of (2.28). The equation (2.30) is the estimation of (2.27). And the amplitude spectrum $S(\omega)$ is estimated by (2.31). The weight vector $\boldsymbol{W}$ is calculated by (2.32).

$$\boldsymbol{R}^i = \frac{\boldsymbol{1}}{\boldsymbol{N}} \boldsymbol{E}\boldsymbol{W}^i\boldsymbol{E}^H \tag{2.29}$$

$$\boldsymbol{F}^i = \boldsymbol{u}\boldsymbol{A}^i = \boldsymbol{u}(\boldsymbol{R}^i)^{-1}\boldsymbol{E}\boldsymbol{W}^i \tag{2.30}$$

$$\boldsymbol{S}^i = \frac{\boldsymbol{u}(\boldsymbol{R}^i)^{-1}\boldsymbol{E}}{diag(\boldsymbol{E}^H\boldsymbol{R}^i\boldsymbol{E})} \tag{2.31}$$

$$\boldsymbol{W}^{i+1} = diag(\left|\boldsymbol{S}^i\right|^2) \tag{2.32}$$

where superscript i is iteration number and $\boldsymbol{W}^1 = \boldsymbol{I}$ for the first iteration. The superscript H stands for Hermitian transpose. $\boldsymbol{E}(K \times N)$ has elements $e^{-j\omega_n kT}$that is a vector form of $\boldsymbol{E}_\omega$ where continuous frequency ω is estimated by using ω_nThe above iteration stops if the result of $\boldsymbol{S}^i$ at the current iteration is close to the last iteration.

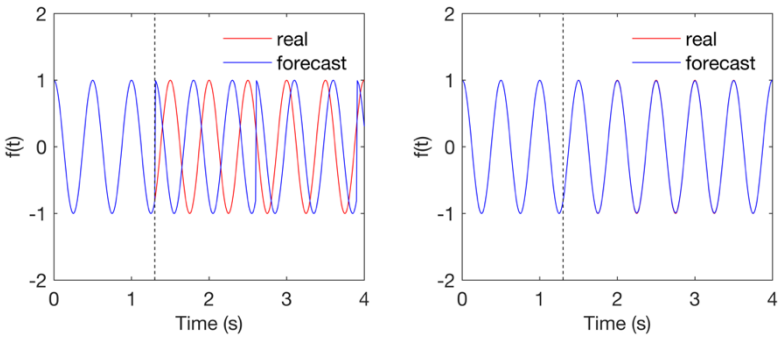
Now, the output of the EDFT module $\boldsymbol{u}_e$ can be extrapolated. As the number of frequency component N is larger than the number of input data K , by using the inverse discrete Fourier transform (IDFT) (2.33), we can obtain a time sequence $\boldsymbol{u}_e$ that contains a number of N-K extrapolated data.

$$\boldsymbol{u}_e = \frac{1}{N}\boldsymbol{F}\boldsymbol{E}_N^H \tag{2.33}$$

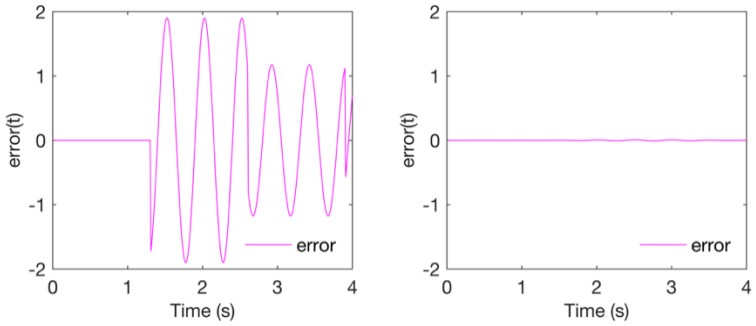
where $\boldsymbol{E}_N(N \times N)$ is a matrix that has element $e^{-j2\pi f_n mT}, m = 0,1 \ldots, N-1$ and $\omega_n = 2\pi f_n$.

Example 2: In order to show the ability of the proposed EDFT method to perform forecast, a simple example is shown below. A cosine wave with amplitude 1 and frequency 2 Hz (e.g., mode 1 in Example 1) is used as the input and the sampling frequency is 100 Hz . The observation time period is 0 - 1.3 seconds and the data within the time interval 1.3 - 4 seconds need to be forecasted. In other words, the first 2.6 cycles are trucked as measured data and we need to forecast the following 5.4 cycles data. In this example, we use the DFT as a comparison and the number of frequency points $N = 4{,}000$ is selected which is larger than the number of observation data $K = 1{,}300$. Figure 2.6 (a) shows the forecast results derived from the DFT and

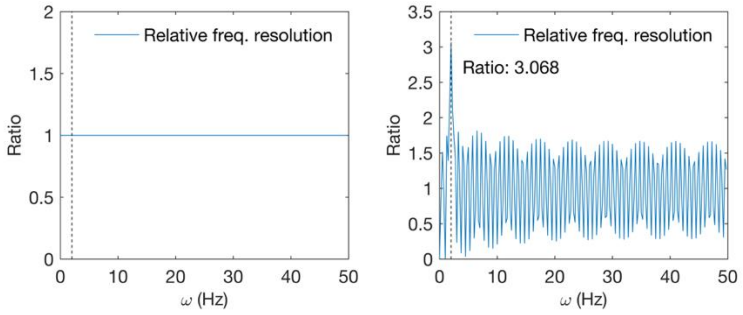
EDFT methods, respectively. Their forecast errors in time domain are shown in Figure 2.6(b). It is obvious that the DFT method fails to provide accurate forecasts. The DFT predictions between time interval 1.3-2.6 seconds and 2.6-3.9 seconds are the same as data within the observation period 0-1.3 seconds. However, the EDFT method can provide more accurate forecast results. We also analyze the performance of the DFT and EDFT methods in frequency domain.



(a) The DFT (left) and EDFT (right) forecasts



(b) The DFT (left) and EDFT (right) forecast errors



(c) Weight of the DFT (left) and EDFT (right) put on frequency components

Figure 2.6 Example of DFT and EDFT methods

To identify periodicities within the input data, algorithms should put high resolution around the true frequency components. The frequency resolution of the DFT method is $1/K$ and the highest frequency resolution of the EDFT method can reach$1/N$. Therefore, the EDFT method's frequency resolution is N/K times higher than the DFT method. Besides, in order to emphasize true frequency, algorithms should put more weight on it. For the DFT method, the weight on every frequency component is $\boldsymbol{w} = diag(\frac{1}{N}\boldsymbol{E^H E})$ which equals to K/N. On the other hand, the EDFT method puts $\boldsymbol{w_a} = diag(\frac{1}{N}\boldsymbol{E^H A})$ weight on every frequency component and w_a is within range $[0, 1]$. Compare with two weights illustrated in Figure 2.6(c), to some frequencies, the weight of the EDFT method $\boldsymbol{w_a}$ is N/K times larger than the weight of the DFT method $\boldsymbol{w}$. We can tell that the weight by the EDFT method at frequency $2Hz$ is $N/K = 4{,}000/1{,}300 = 3$ times more than the weight by the DFT method.

2.2.3 Price spike oversampling module

As the low occurrence rate of price spike affects classifiers (e.g., SVM) to recognize the pattern of the spike, the price spike oversampling module is utilized in our proposed forecast framework to increase the number of price spike cases. Price spike cases can be divided into two parts: predictor variables and target values. The ESPO method [25] is used to oversample predictors while preserving temporal relationship among data and the SMOTE for regression method [26], [27] is used to oversample the targets.

To create the predictors of spike cases using the ESPO method, we firstly calculate the spike case predictors covariant matrix $\boldsymbol{W}$ by using (2.34).

$$\boldsymbol{W} = \frac{1}{N_{case}} \sum_{i=1}^{N_{case}} (\boldsymbol{P}_i - \overline{\boldsymbol{P}})(\boldsymbol{P}_i - \overline{\boldsymbol{P}})^T \qquad (2.34)$$

where $\overline{\boldsymbol{P}} = 1/N_{case} \sum_{i=1}^{N_{case}} \boldsymbol{P}_i$ represents the mean of spike case predictors and $\boldsymbol{W}$ is a positive semi-definite covariant matrix. Then, we utilize the eigen decomposition (2.35) on $\boldsymbol{W}$.

$$\boldsymbol{D} = \boldsymbol{V}^T \boldsymbol{W} \boldsymbol{V} \tag{2.35}$$

where $\boldsymbol{D}$ is the diagonal eigenvalue matrix with descending order; the $\boldsymbol{V}$ is the eigenvector matrix of $\boldsymbol{W}$; and the superscript T means transposition. Eigenvectors of a covariance matrix can be viewed as data varying directions and eigenvalues are the magnitude of variance on the direction. In order to create new spike cases that have a similar temporal structure as the real spike cases, in the eigenvector directions derived from equation (2.35), the eigenvalues of synthetic cases should be comparable with the eigenvalues of real cases.

Although the eigenvalue matrix $\boldsymbol{D}$ can be measured from data, to facilitate data synthesis, it is convenient to scale the eigenvector matrix $\boldsymbol{V}$ and convert $\boldsymbol{D}$ to an identity matrix $\boldsymbol{I}$. Therefore, we scale the eigenvector matrix $\boldsymbol{V}$ using (2.36) to divide each eigenvector v_k by $\sqrt{d_k}$, where d_k is the eigenvalue of the eigenvector v_k.

$$\boldsymbol{F} = \boldsymbol{V}\boldsymbol{D}^{-1/2} \tag{2.36}$$

Then, the equation (2.35) can be rewritten as (2.37).

$$\begin{aligned} \boldsymbol{I} &= \boldsymbol{D}^{-\frac{1}{2}} \boldsymbol{V}^T \boldsymbol{W} \boldsymbol{V} \boldsymbol{D}^{-\frac{1}{2}} = \boldsymbol{F}^T \boldsymbol{W} \boldsymbol{F} \\ &= \frac{1}{N_{case}} \sum_{i=1}^{N_{case}} \boldsymbol{F}^T (\boldsymbol{P}_i - \overline{\boldsymbol{P}})(\boldsymbol{P}_i - \overline{\boldsymbol{P}})^T \boldsymbol{F} \\ &= \frac{1}{N_{case}} \sum_{i=1}^{N_{case}} (\boldsymbol{F}^T \boldsymbol{P}_i - \boldsymbol{F}^T \overline{\boldsymbol{P}})(\boldsymbol{F}^T \boldsymbol{P}_i - \boldsymbol{F}^T \overline{\boldsymbol{P}})^T \end{aligned} \tag{2.37}$$

If we define

$$\boldsymbol{Z} = \boldsymbol{F}^T \left(\widetilde{\boldsymbol{P}} - \overline{\boldsymbol{P}} \right) \tag{2.38}$$

$\widetilde{\boldsymbol{P}}$ are the synthesized predictors. We generate $\boldsymbol{Z}$ from multivariate Gaussian distribution $N(\boldsymbol{0}, \boldsymbol{I})$ as a large number of random and independent distributions that obey the Gaussian distribution

[29]. Therefore, if we generate a sufficient number of rare cases, after eigen decomposition and scaling, the eigenvalue matrix of the generated cases covariance matrix also has an identify matrix structure, which ensures the synthesized predictors have the same temporal structure as the original rare cases. After getting $\boldsymbol{Z}$, the synthesized predictors $\widetilde{\boldsymbol{P}}$ can be derived from (2.39).

$$\widetilde{\boldsymbol{P}} = \boldsymbol{V}\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{Z} + \overline{\boldsymbol{P}} \tag{2.39}$$

After synthesizing predictors, the next step is to synthesize the target value of spike case. The SMOTE for regression method [27], as a weighted average method where the larger distance has a smaller weight, is used in this step to calculate the values of the synthesized target. First, a spike case is selected with the synthesized predictor derived by using (2.39) and its R nearest spike cases are identified in the real spike case dataset. Then, the distance between the predictor and its R nearest spike cases are calculated. Finally, the equation (2.40) is adopted to calculate the value of its target G_s.

$$G_s = \frac{\sum_{r=1}^{R} G_r L_{rs}}{\sum_{r=1}^{R} L_{rs}} \tag{2.40}$$

L_{rs} is the Euclidean distance between predictor values of real case r and synthetic case s. G_r is the target values of the R nearest spike cases in the real spike case dataset.

Example 3: The following example shows that when the number of interested cases is limited, using the oversampled dataset could provide a better forecast result compared with only using the original dataset in [28]. The training dataset contains 300 cases and each case has 60 predictors and 1 target. As listed in Table 2.1, the cases with the target value 1 are rare cases and account for $1/6$ of the original dataset. After oversampling by the proposed methods, the number of rare cases increases to 250 and accounts for 50% of the new dataset.

Table 2.1 Information of dataset

Dataset	Rare cases (R)	Normal cases (N)	R/N
Original	50	250	20%
Oversampling	250	250	50%

Table 2.2 Confusion matrix derived from the SVM trained by the original dataset

Items	True Rare Target	True Normal Target
Rare Prediction	5	0
Normal Prediction	5	10

Table 2.3 Confusion matrix derived from the SVM trained by the oversampled dataset

Items	True Rare Target	True Normal Target
Rare Prediction	9	1
Normal Prediction	1	9

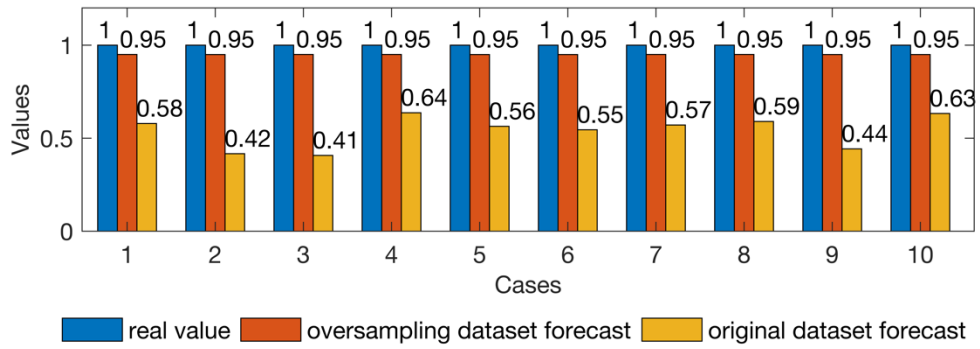


Figure 2.7 Results of forecast using original and oversampled dataset

There are 20 cases selected to be tested, in which 10 cases have normal targets and the other 10 cases have rare targets. In order to forecast the target value of the rare cases, firstly, we need to identify whether a case is a rare case or not. In Table 2.2 and Table 2.3, we list the classification results of the SVM classifier trained by the original dataset and the oversampled dataset, respectively. From Table 2.2, we can see that there are 5 rare targets are incorrectly classified as normal targets and only 5 rare targets are identified. However, from Table 2.3, the classifier trained by using the oversampled dataset almost identified all rare targets. Only 1 rare

target is incorrectly classified as a normal target. Then, the rare target numeric forecast results are demonstrated in Figure 2.7. we can tell that the SVM trained by the oversampled dataset for the regression has higher forecast accuracy than the one trained by the dataset without being oversampled.

2.3 Conclusion

The proposed forecast framework considered both periodic and volatile characteristics of short-term electricity prices. In this chapter, three key methodologies and their potentials were presented in detail. The VMD module can separate the price data into several modes and be able to remove more random noises from the original data. Without assuming the data outside the observation period is the same as the data within the period, the EDFT module can achieve a higher resolution in frequency domain. The oversampling module with ESPO and SMOTE for regression methods can properly synthesize more price spike cases and enhance the training dataset. All of those key modules are properly integrated into our proposed forecast framework to provide an accurate and reliable price prediction. Further, the performance of the proposed price forecast framework and its key modules will be tested on several major electricity markets including PJM, MISO, ISO-NE, NYISO, et cetera. The detailed results will be discussed in the next chapter.

CHAPTER III

PRELIMINARY RESULTS OF POINT FORECAST

This chapter is used to evaluate the effectiveness of the proposed price forecast framework. In this chapter, we present forecast results by using the proposed short-term electricity price forecast framework. Multiple electricity markets are studied and proper evaluation metrics are selected to estimate the performance of key modules of the proposed short-term price forecast framework. Numerous results illustrate that the proposed forecast method is capable to accurately predict both normal and spike electricity prices. In this chapter, the electricity markets that are used to test our proposed forecast method are presented in section 3.1. In order to evaluate the classification and regression forecast results, we introduce evaluation metrics in section 3.2. The normal electricity price forecast and price spike prediction results are shown in section 3.3 and 3.4, respectively. Section 3.5 demonstrates more forecast results by using data from different seasons of several major markets. Conclusions are drawn in section 3.6.

3.1 Introduction

Extensive researches on the short-term electricity price forecast have been conducted to improve its accuracy. As normal electricity prices can be viewed as time series data and demonstrate periodic patterns, researchers have put a lot of effort into exploring periodicities within electricity prices in time domain which is a type of trial-and-error process. However, if price data are transferred into frequency domain, we are able to explore more price periodicities which should be able to provide more accurate forecasts. Besides, as price spikes rarely

occurred, existing forecast methods are hard to learn patterns from limited data and difficult to provide high-quality predictions. Therefore, the accuracy of price spike forecast could improve if more spikes data exist in the price dataset.

In Chapter 2, a new forecast framework was proposed to predict short-term electricity prices. To accurately estimate price periodicity and reduce random noise, the variational mode decomposition (VMD) method is firstly utilized to decompose electricity prices into multiple modes. Then, the extended discrete Fourier transform (EDFT) method is applied to perform normal price forecast for each mode as it can achieve a high resolution around dominant frequencies within price modes and does not simply assume that data outside the observation period repeat data within that period. In addition, to enhance the price spike forecast, data oversampling methods such as the enhanced structure preserving oversampling (ESPO) method and the synthetic minority oversampling technique (SMOTE) for regression method are applied to synthesize spikes data.

In this chapter, numerical case studies are presented and analyzed to demonstrate the effectiveness of the proposed short-term electricity price forecast framework as well as the performance of the used methodologies for both normal and spike electricity price forecasts by using the following price data sources:

- Pennsylvania-New Jersey-Maryland Interconnection (PJM) [30]
- Midcontinent Independent System Operator (MISO) [31]
- ISO New England (ISO-NE) [32]
- New York Independent System Operator (NYISO) [33]
- Southwest Power Pool (SPP) [34]
- Day ahead electricity market of mainland Spain [35]

3.2 Evaluation metrics

In order to evaluate the performance of the proposed short-term price forecast framework, metrics should be carefully selected as different forecast targets have distinctive assessment requirements. For numerical forecast results, we commonly utilize the mean absolute percentage error (MAPE) metric to assess forecast accuracy. In equation (3.1), the P_t and F_t represent the predicted and real electricity prices, respectively. The T is the forecast time period and the subscript t is the measured time point. Besides, we utilize the variance of forecast errors to measure uncertainty. The smaller the variance, the less uncertain is the model or the more accurate are the forecast results. The variance of the forecast errors is calculated by using equation (3.2) where the σ^2 represents variance and σ is the standard deviation of errors. Although the MAPE is a common metric to evaluate the accuracy of forecast results, under certain circumstances, the MAPE metric may not work well. For instance, if one electricity price is close to zero, the value of MAPE tends to be infinite and cannot fairly demonstrate the effectiveness of forecast methods. Therefore, the average-mean absolute percentage error (AMAPE) metric (3.3) is also applied in which the dominator is $\overline{P_t} = 1/T \sum_{t=1}^{T} P_t$ as the probability that all electricity prices within the forecast period are almost zero is very low.

$$MAPE = \left[\frac{1}{T}\sum_{t=1}^{T} \frac{|P_t - F_t|}{P_t}\right] \times 100\% \tag{3.1}$$

$$\sigma^2 = \frac{1}{T}\sum_{t=1}^{T} \left[\frac{|P_t - F_t|}{\overline{P_t}} - MAPE\right]^2 \tag{3.2}$$

$$AMAPE = \left[\frac{1}{T}\sum_{t=1}^{T} \frac{|P_t - F_t|}{\overline{P_t}}\right] \times 100\% \tag{3.3}$$

On the other hand, unlike normal electricity prices, price spikes have low occurrence rates and large magnitude values. Therefore, for the price spike forecast, we need to evaluate

whether or not the forecast methods can identify price spikes. To evaluate such classification performance, metrics such as precision and recall are commonly used. Precision (3.4) is the ratio of the number of correctly predicted spikes to the number of total predicted spikes. Recall (3.5) is the ratio of the number of correctly predicted spikes to the number of total true spikes. The symbols used in (3.4) and (3.5) are defined in the confusion matrix in Table 3.1.

$$Precision = \frac{TS}{TS + FS} \tag{3.4}$$

$$Recall = \frac{TS}{TS + FN} \tag{3.5}$$

Table 3.1 Confusion matrix

Items	**True Spike**	**True Normal**
Spike Prediction	True Spike (TS)	False Spike (FS)
Normal Prediction	False Normal (FN)	True Normal (TN)

For price spike forecast, we care more about recall than precision as it is vital to classify as many true price spikes as possible. Therefore, we also utilize the F2-score to evaluate the price spike classification performance. The F2-score in (3.6) not only combines both precision and recall measurements into a single value but also focuses more on recall than precision.

$$F2 - score = \frac{5 \times Precision \times Recall}{4 \times Precision + Recall} \tag{3.6}$$

3.3 Case 1: normal electricity price forecast

In this section, the proposed normal price forecast is tested by using the PJM price dataset. Five days day-ahead electricity prices are chosen to be forecasted. The chosen dates include January 20, February 10, March 5, April 7, and May 13 in the year 2006. Those dates have also been used in [12], [10] to test their price forecast methods. The comparisons between

our forecast results and those in the published papers are discussed in 3.3.1. Besides, the detailed results of key forecast modules for the normal price prediction are explained in 3.3.2

3.3.1 Forecast results of the PJM market

In this subsection, to measure the accuracy of the price forecast, the AMAPE value of our proposed normal price forecast method is calculated and compared with that of the published forecast methods in [12] and [10]. The comparison results are listed in Table 3.2. The first column indicates the forecast date. The AMAPE values of our proposed forecast method, [12], and [10] are listed in the column 2, 3, and 5, respectively. A lower AMAPE value means a more accurate forecast result. Columns 4 and 6 list the forecast accuracy improvements of our forecast method compared with [12] and [10], respectively. A positive value means that the price predicted by our forecast method is more accurate than the results reported in the references. From Table 3.2, we can tell that the AMAPE values of our proposed method are around 3%. Compared with [10], our method has an overall better performance as the AMAPE values of [10] are basically above 6%. The forecast accuracy of our method is also normally better than [12] except for the price prediction for February 10. However, the AMAPE value of our method on February 10 is 3.12% which is still acceptable and close to the AMAPE value of 2.85% in [12]. The largest AMAPE improvement of our proposed method compared with [12] and [10] are 53.8% and 69.7% on March 5, respectively.

Table 3.3 shows the variance of forecast errors, σ^2. A smaller variance means a more stable forecast result. As we can see, the variance of the proposed forecast method is between 0.00043 on March 5 and 0.00078 on February 10. Compared with those in [12] and [10], the variance of forecast errors of the proposed method is overall lower. The largest variance

improvements of our method compared with [12] and [10] are 87% and 93% on March 5, respectively.

Table 3.2 Comparison of daily AMAPEs for case 1

Date of 2006	Proposed	Ref. [12]	Improv.	Ref. [10]	Improv.
Jan. 20	2.61%	3.71%	29.6%	6.93%	62.3%
Feb. 10	3.12%	2.85%	-9.47%	7.96%	58.9%
Mar. 5	2.53%	5.48%	53.8%	7.88%	67.9%
Apr. 7	3.15%	4.17%	24.5%	9.02%	65.1%
May 13	3.27%	4.06%	20.5%	6.91%	52.7%

Table 3.3 Comparison of daily error variances for case 1

Date of 2006	Proposed	Ref. [7]	Improv.	Ref. [8]	Improv.
Jan 20	0.00070	0.0010	30.0%	0.0010	79.4%
Feb 10	0.00078	0.0050	84.4%	0.0050	84.4%
Mar 5	0.00043	0.0033	87.0%	0.0033	93.0%
Apr 7	0.00051	0.0013	60.8%	0.0013	86.6%
May 13	0.00043	0.0015	71.3%	0.0015	91.2%

Figure 3.1-3.5 compare the forecasted and real electricity prices and further illustrate the accuracy of our electricity price forecasts for the chosen days of January 20, February 10, March 5, April 7, and May 13, 2006, respectively. From Figure 3.1-3.3, we can see that there are two price peaks within each day: morning and evening peaks. However, on March 5, the peak pattern changes and shows in Figure 3.3 that the value of the evening peak is higher than the morning peak. As the methods applied in [7] and [8] rely on recognizing repetitive patterns within historical data and utilizing such patterns to perform forecast, when price daily pattern changed, their forecast accuracy may decrease, which was supported by the numerical results listed in Table 3.2 and 3.3. From Figure 3.4 and 3.5, we can tell that there are two peak periods within a day and the peak values are almost the same. Therefore, the forecast performance of our method

for April 7 and May 13 are alike that the AMAPE for those two days are 3.15% and 3.27% and the variances of forecast errors are 0.00051 and 0.00043, respectively

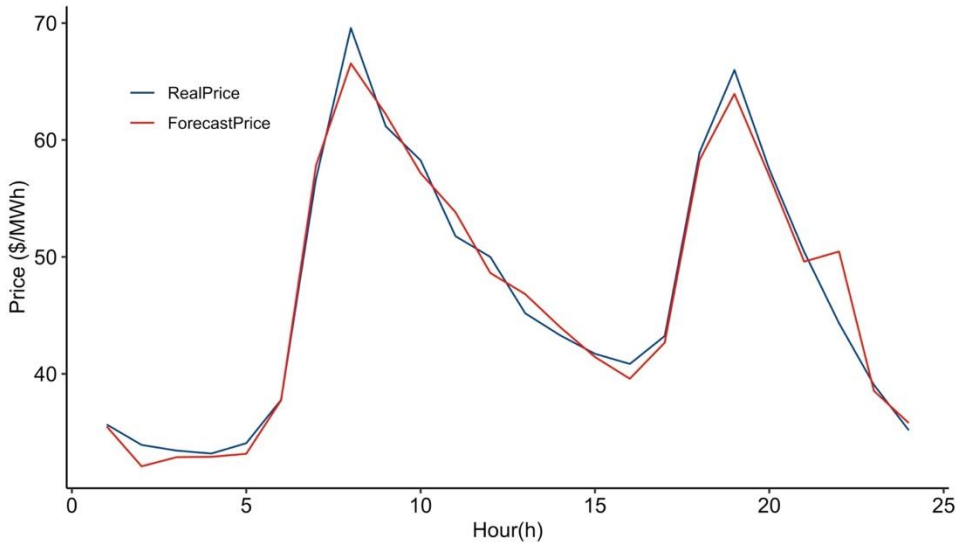


Figure 3.1 Real and forecasted PJM electricity price (January 20, 2006).

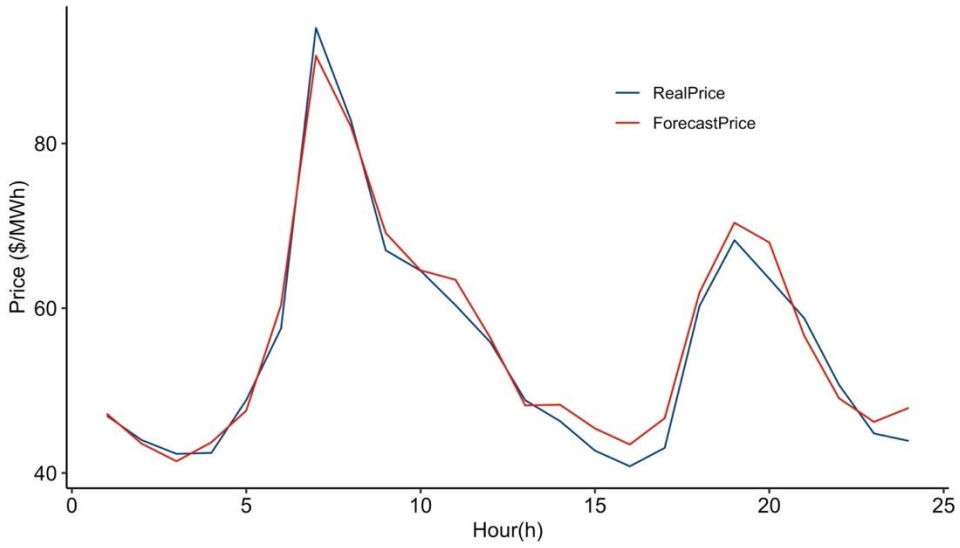


Figure 3.2 Real and forecasted PJM electricity price (February 10, 2006).

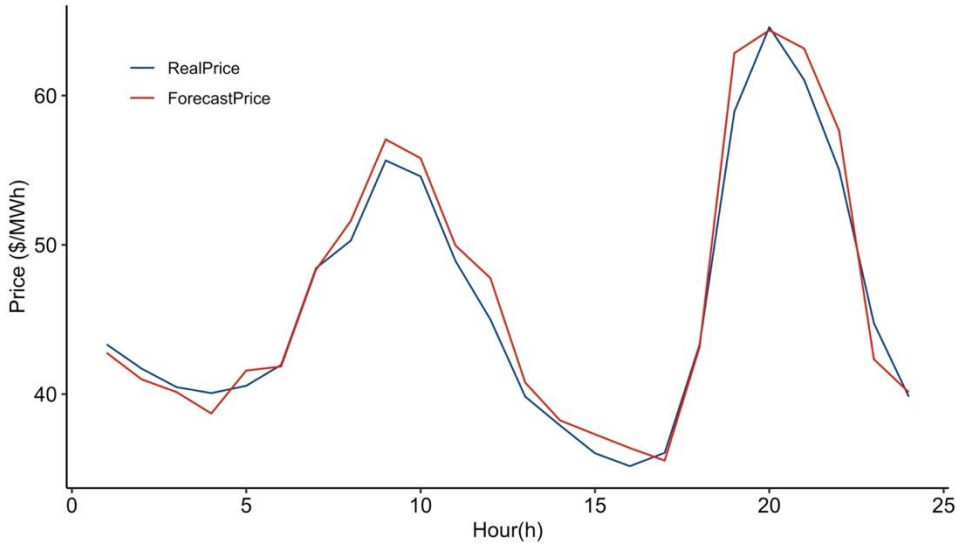


Figure 3.3 Real and forecasted PJM electricity price (March 5, 2006).

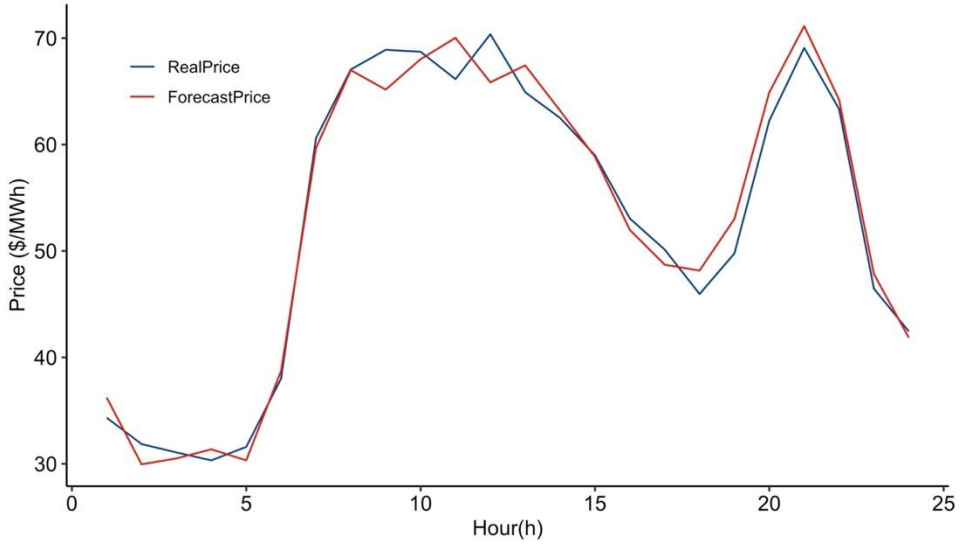


Figure 3.4 Real and forecasted PJM electricity price (April 7, 2006).

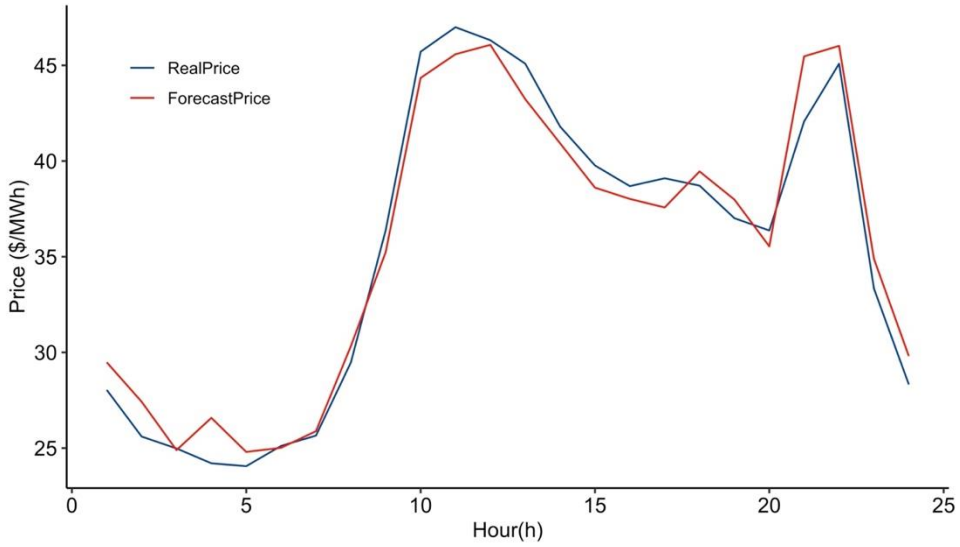


Figure 3.5 Real and forecasted PJM electricity price (May 13, 2006).

3.3.2 Detailed results of each key module

To better understand key forecast modules for the normal price prediction, electricity prices on March 5, 2006 are selected to show the results of each key forecast module. According to the proposed price forecast framework, the first step is to read data and perform data preprocessing. After loading the PJM price dataset, no missing or duplicated price data are to be found. The imported historical price data from December 1, 2005 to March 4, 2006 are shown in Figure 3.6.

The second step of our proposed forecast procedure is to perform the time series separation. As the variance of electricity prices doesn't increase with time, we utilize an additive time series decomposition model to separate trend, seasonal, and remaining data, which are shown in Figure 3.7.

As the price data components derived from step 2 are still mixed with frequency components, the third step of our proposed forecast procedure is to perform the VMD

decomposition to transform each price data component into multiple modes where each mode is band-limited data with a center frequency. We separate the trend data into four modes, the seasonal data into one mode, and the remaining data into four modes. As an example, the periodogram of the remaining data is shown in Figure 3.8, which contains four major frequencies marked by dotted circles. Accordingly, the decomposed four modes of the remaining data are illustrated in Figure 3.9.

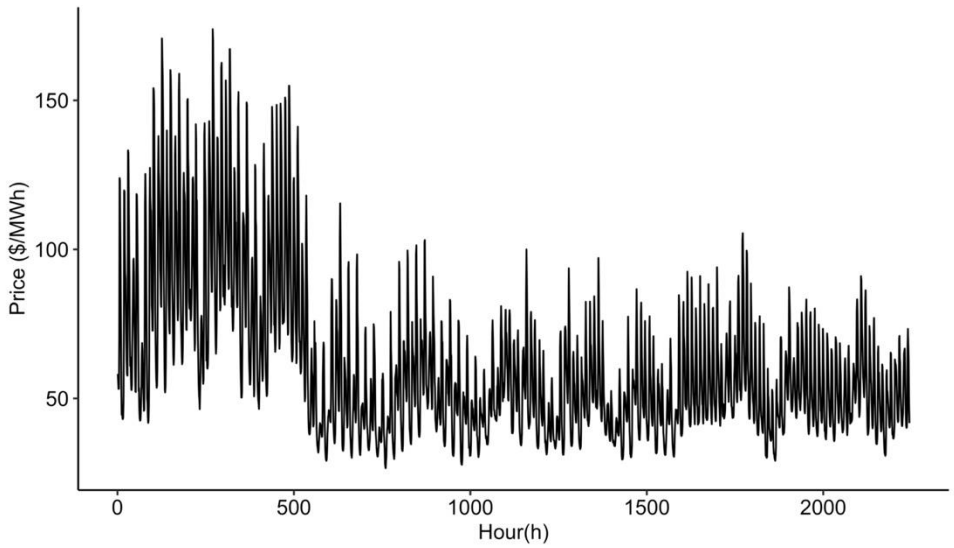


Figure 3.6 PJM electricity price for the period of December 1, 2005 to March 4, 2006.

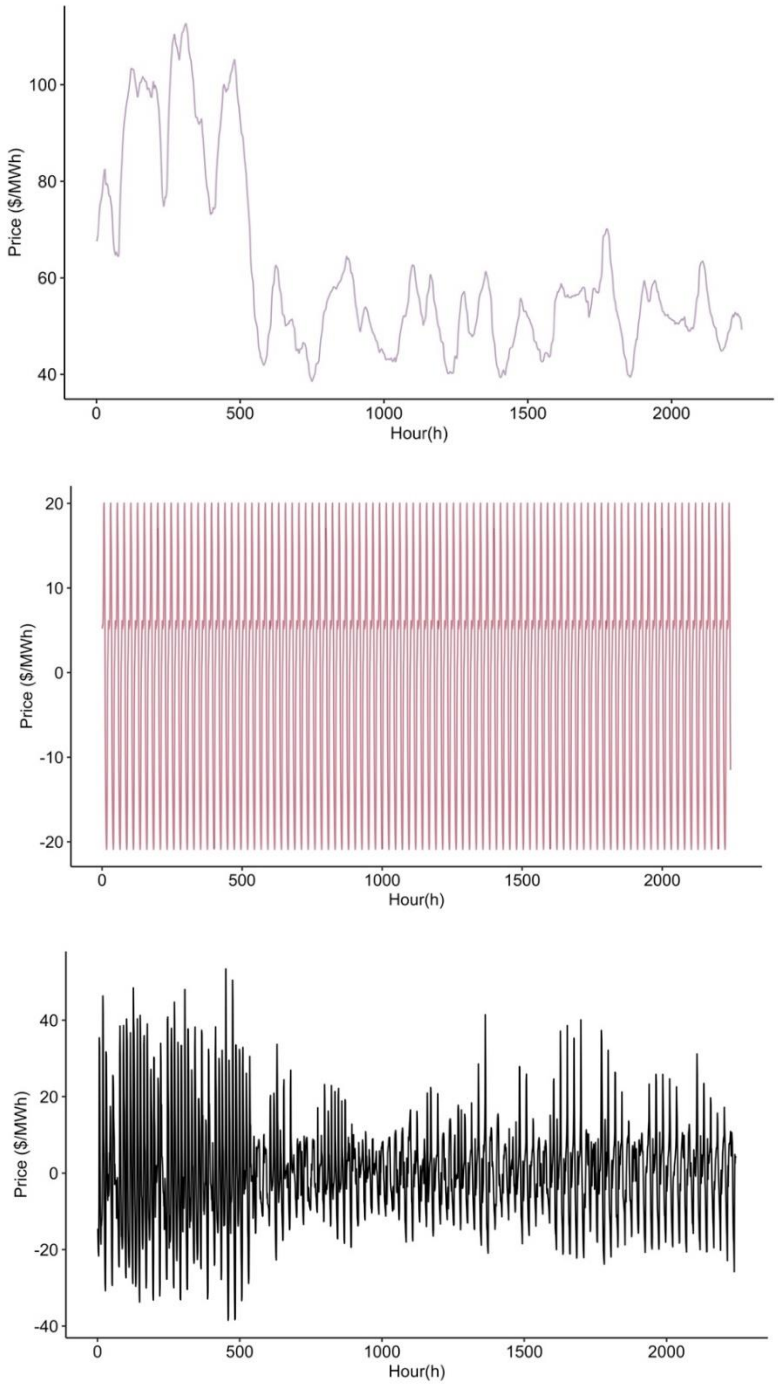


Figure 3.7 Time series decomposition of the imported PJM electricity price for the period December 1, 2005 to March 4, 2006. Trend (top), seasonal (middle), and the remaining data (bottom).

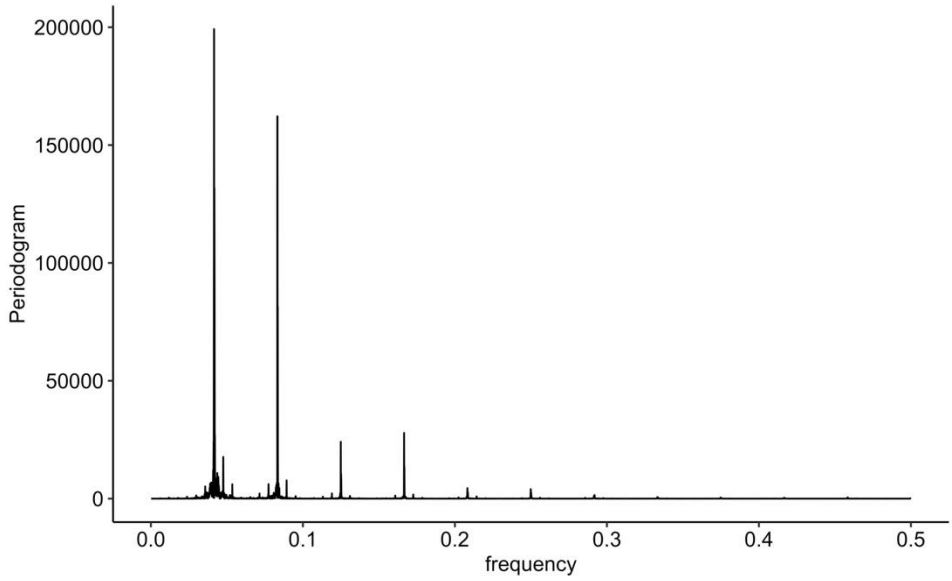


Figure 3.8 Periodogram of the remaining data of the PJM electricity price for the period December 1, 2005 to March 4, 2006.

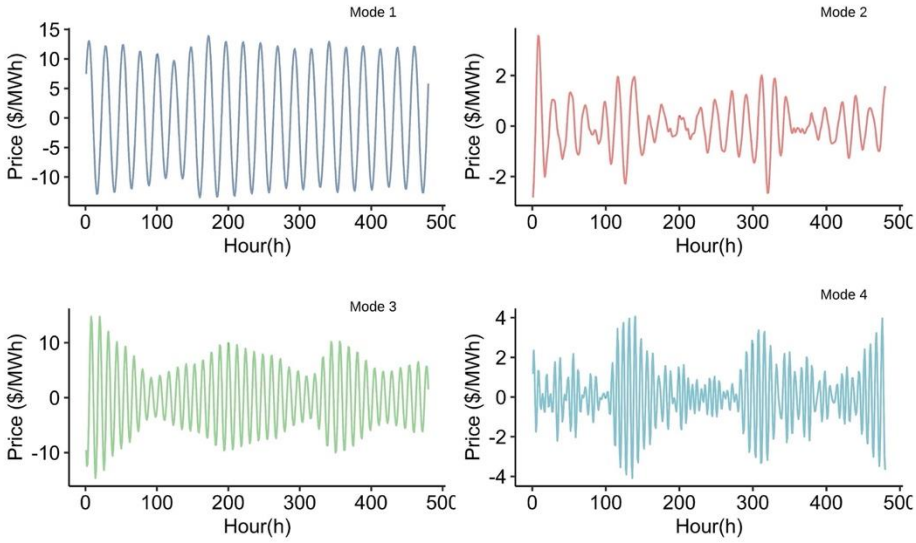


Figure 3.9 VMD decomposed remaining data of the PJM electricity price for the period December 1, 2005 to March 4, 2006.

The fourth step is to utilize the EDFT method to perform the forecast on the decomposed price modes. Figure 3.10 shows an example of the modes forecast for the remaining data.

Obviously, the forecasted modes, except for mode 2, perfectly trace their real values. Although the forecast accuracy of mode 2 of the remaining data is not good enough, it doesn't have a significant impact on the price forecast accuracy as it is only a tiny portion of the total price.

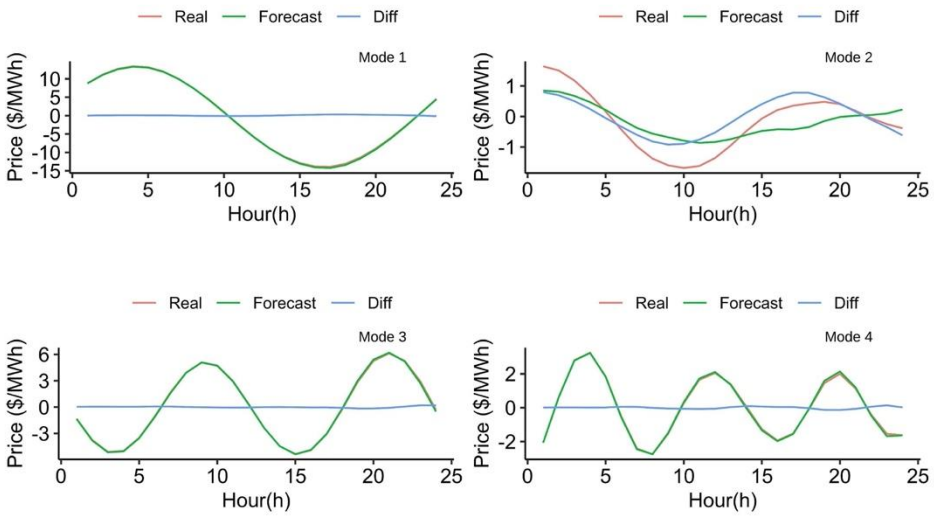


Figure 3.10 EDFT forecast for the remaining data of the PJM electricity price on March 5, 2006.

The final step is to compose the forecast results. Based on the additive model assumption, forecast results of the trend, seasonal, and remaining data/modes are added together to obtain the final normal price prediction result which is shown in Figure 3.11. Finally, the forecasted price is very close to the real one which has been shown in Figure 3.3. The reference [10] mentioned that the peak hour electricity price is hard to predict and may worsen the performance of the total forecast. However, by applying our proposed forecast modules, the peak price prediction at hour 20 is more accurate and the AMAPE 2.53% of our forecast is lower than the AMAPE values reported in [12] and [10] that are 5.48% and 7.88%, respectively.

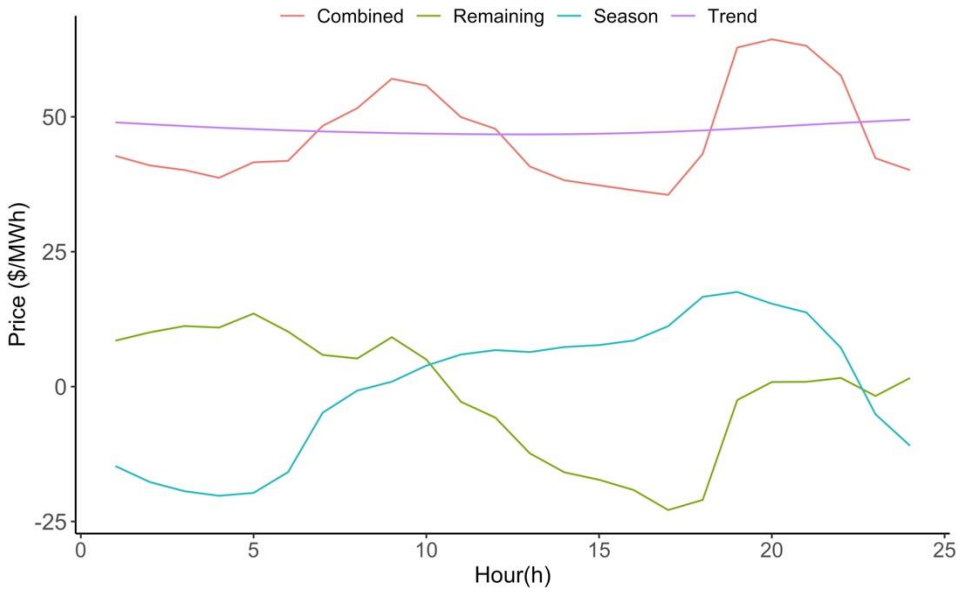


Figure 3.11 The trend, seasonal, remaining, and combined day-ahead electricity price forecast of the PJM day-ahead market on March 5, 2006.

3.4 Case study 2: electricity price spike forecast

The electricity price spike is a type of abnormal data within the day-ahead electricity price dataset as it rarely happens and is larger than the normal electricity price. Due to low occurrence rate, price spike is hard to be identified by normal classification algorithms as those methods assume a balanced dataset that means there are almost equal amount of data from different types. Besides, as classifiers are hard to identify price spikes from an imbalanced dataset, the regression algorithms are barely able to provide accurate price spike values. To address those problems, we use oversampling methods to increase the number of price spikes to facilitate classification algorithms. Once classification algorithms could identify price spikes correctly, regression algorithms that are specifically tuned will provide more precise quantity predictions on price spikes. In the following two subsections, the performances of classification and regression forecasts are discussed.

3.4.1 Price spike classification results

Before utilizing oversampling methods, firstly, we need to define what is price spike. Different researchers utilized different price spike definitions in their studies. According to [14] and [17], price spikes are electricity prices over the threshold value $\mu + 2\sigma$, where μ is the mean value of the historical data set and σ is the standard deviation of historical prices. Besides, [19] defined that electricity prices above 150 $/MWh or 200 $/MWh are price spikes. Noticed that the $\mu + 2\sigma$ definition will be heavily influenced by several large electricity prices as the mean value is easily affected by extreme values. And the thresholds such as 150 $/MWh or 200 $/MWh are too rigid and may not be true for certain markets or circumstances. Therefore, we decide to use the interquartile range (IQR) method to identify price spikes, which means electricity prices are defined as price spikes if their values are higher than $\mathrm{Q}_3 + 1.5 \times \mathrm{IQR}$. The Q_3 is the third quartile and the IQR is the difference between the third quantile Q_3 and the first quantile Q_1 that is $IQR = Q_3 - Q_1$. One of the benefits of using the IQR method is that the quantile is not affected by extreme values as the quantiles are called resistant measures. The other benefit is that it is more flexible than a hard threshold.

Figure 3.12 shows the hourly means plot for day-ahead electricity prices of PJM from January 2, 2005 to July 31, 2006. According to Figure 3.12, it is obvious that day-ahead electricity prices at different hours have different patterns. Therefore, it is unfair to use a single threshold to tell whether an electricity price is a price spike or not. If we utilize the same threshold to define a price spike, price spikes at hour 3 might not be treated as abnormal electricity prices compared with the prices from on-peak hours 7 a.m. - 11 p.m. [30], but they are actually unusual data occurred at hour 3. So, we utilize the IQR method to identify price spikes for each hour.

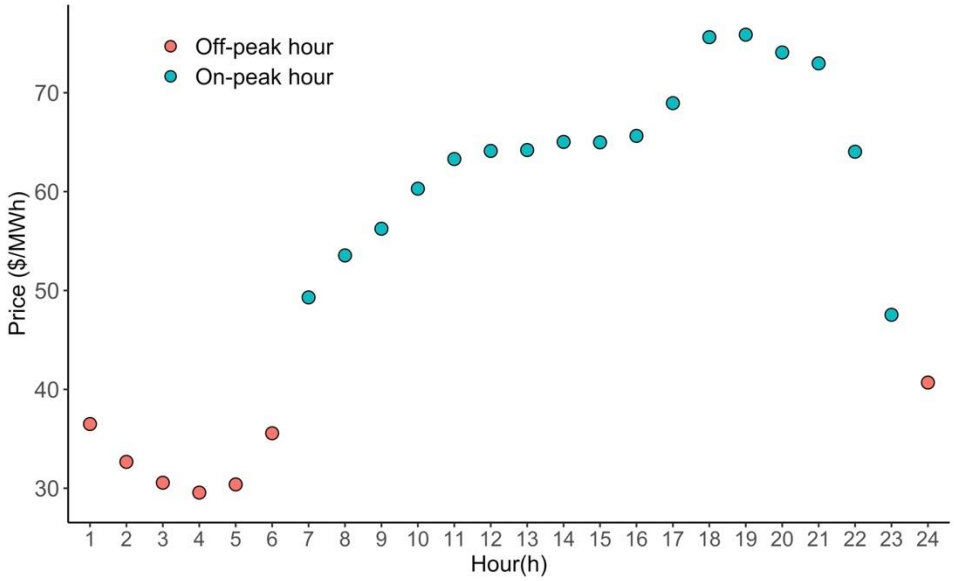


Figure 3.12 Hourly means plot for day-ahead electricity prices of PJM from January 2, 2005 to July 31, 2006.

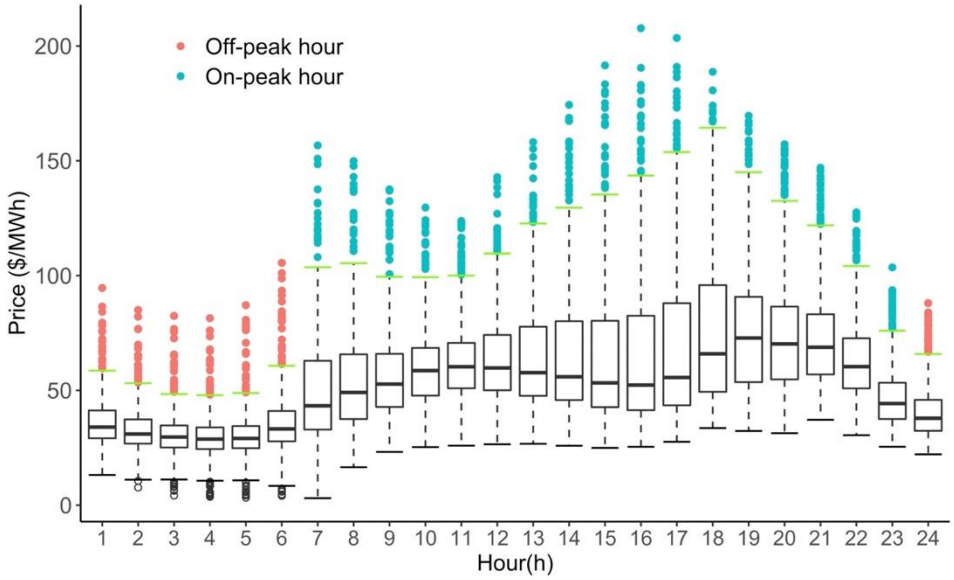


Figure 3.13 Hourly price spike boxplot for day-ahead electricity prices from January 2, 2005 to July 31, 2006.

The price spikes identified by using the IQR method are shown in Figure 3.13. The points above the green bar are regarded as price spikes (where the red dots represent price spikes at off-

peak hours 12 p.m. - 6 a.m. [30], and the blue dots are price spikes at on-peak hours). The green bars represent the upper-level threshold for each hour and the value of each green bar is calculated by using the equation $Q_3 + 1.5 \times \text{IQR}$. The boxes demonstrate the electricity price distribution between 25^{th} – 75^{th} percentile for each hour and the range of each box equals to electricity price interquartile range. Besides, the bold black bar in each box is the median of electricity prices for each hour. The median can be used as a measure of central tendency because it is barely affected by extreme values. From Figure 3.13, we can tell that the upper-level thresholds have different values for different hours and the off-peak hour upper-level thresholds are lower than the on-peak hour upper-level thresholds. Besides, compared with the variance of median, the upper-level threshold for each hour fluctuates more heavily. Importantly, in order to perform price spike classification with acceptable accuracy, for different hours, we need to train the classifier with different data.

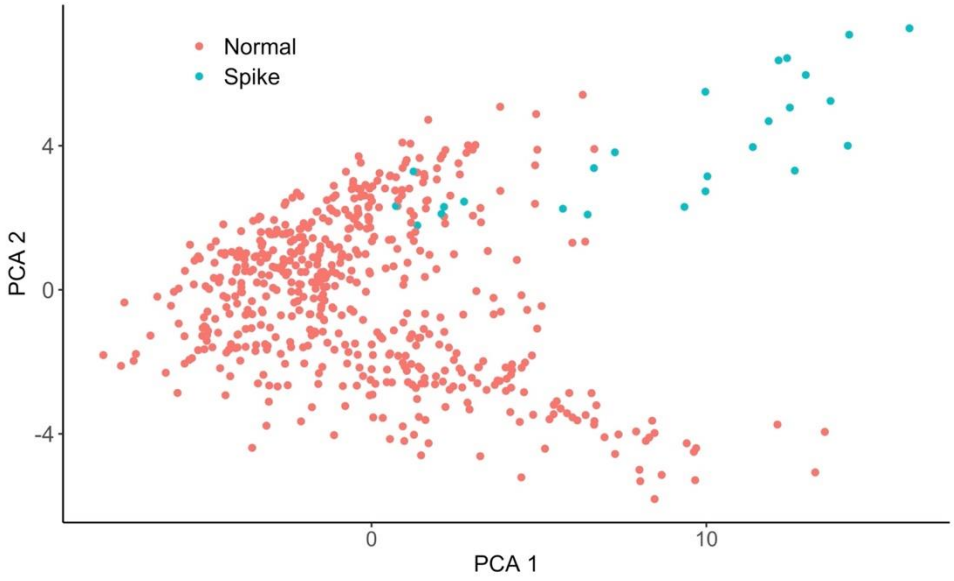
Meanwhile, the statistics related to electricity prices from January 2, 2005 to July 31, 2006 are shown in Table 3.4. Data in the shaded area are related to the on-peak hours and otherwise related to the off-peak hours. The first column indicates the time of a day. The second column shows the upper-level threshold for each hour. If electricity prices are larger than or equal to the threshold, they are regarded as price spikes, otherwise are counted as normal electricity prices. The third and fourth columns show the number of price spikes and normal electricity prices for each hour, respectively. The last column provides the ratio of the number of price spikes to the number of normal prices. Table 3.4 shows that the number of price spikes is far less than the number of normal prices. Even though we summarize almost two years electricity price data, the spike/normal ratios are normally below 10% for each hour.

Table 3.4 Electricity prices statistics from January 2, 2005 to July 31, 2006

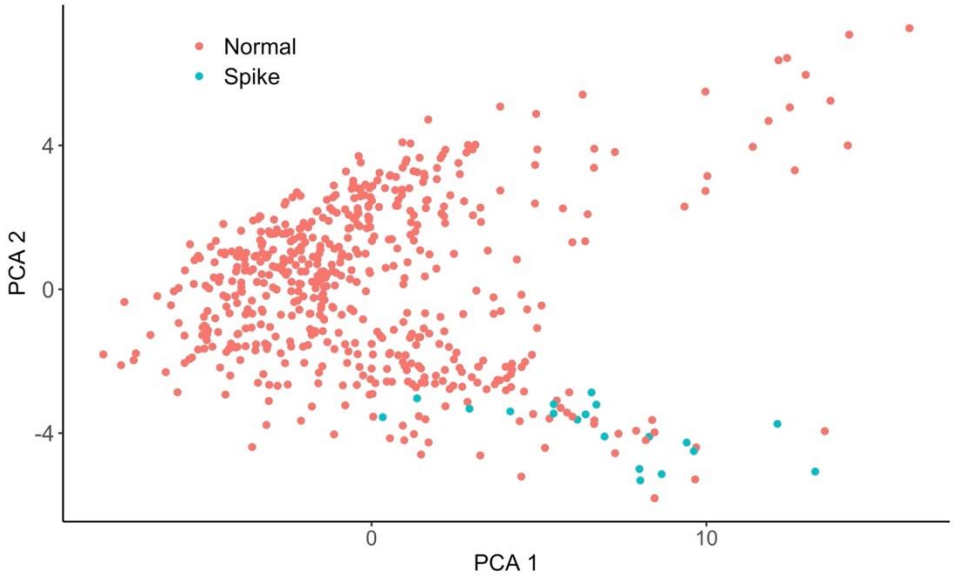
Hour	Threshold ($/MWh)	# of price spikes	# of normal prices	Spike/normal ratio
1	58.60	29	549	5.28%
2	53.15	21	555	3.78%
3	48.40	25	549	4.55%
4	47.92	26	550	4.73%
5	48.85	29	547	5.30%
6	60.72	28	548	5.11%
7	103.69	20	556	3.60%
8	105.43	21	555	3.78%
9	99.48	20	556	3.60%
10	99.30	24	552	4.35%
11	100.03	35	541	6.47%
12	109.61	22	554	3.97%
13	122.72	19	557	3.41%
14	129.57	20	556	3.60%
15	135.30	23	553	4.16%
16	143.57	23	553	4.16%
17	153.82	20	556	3.60%
18	164.43	10	566	1.77%
19	145.07	14	562	2.49%
20	132.54	23	553	4.16%
21	122.48	24	552	4.35%
22	104.16	28	548	5.11%
23	75.98	31	545	5.69%
24	65.85	25	551	4.54%

Therefore, in order to increase the number of price spikes, we adopt the proposed methods that utilize enhanced structure preserving oversampling (ESPO) to oversample price spike classification cases in section 2.2.2 to oversample imbalanced data. Since electricity prices at different hours have different patterns, the oversampling methods are applied to the dataset of each hour. As examples, we oversample the dataset of two hours. One of the selected hours is hour 4 that has the lowest price spike threshold according to Table 3.4 and the other selected hour is hour 17. For the dataset of hour 4, we have oversampled the set of 26 price spike cases to a total of 550 cases that is the amount of normal price cases in the original dataset at hour 4. And,

for the dataset of hour 17, we have oversampled the set of 20 price spike cases to a total of 556 cases. To visualize the effectiveness of our proposed methods, we plot the oversampling results in a two-dimensional feature space. The x-axis and y-axis of Figure 3.14 and 3.15 are the principal components that have the largest and the second largest eigenvalue that are calculated by using the principal component analysis (PCA) method. From Figure 3.14(a), we can see that although the price spike cases in the original dataset are separated from the normal price cases, the total amount of price spike cases is much less than the normal price cases at hour 4. At hour 17 shown in Figure 3.14(b), it is hard to differentiate the price spike cases from the normal price cases as they are mixed together. After oversampling, from Figure 3.15 (a), the number of price spikes at hour 4 increases and the boundary between the normal price cases and the price spike cases is clearer than the original dataset. For hour 17 shown in Figure 3.15(b), after oversampling, although some of the normal price cases are still mixed with the price spike cases, compared with the original dataset, it is much easier to differentiate those two types of cases in the oversampled dataset.

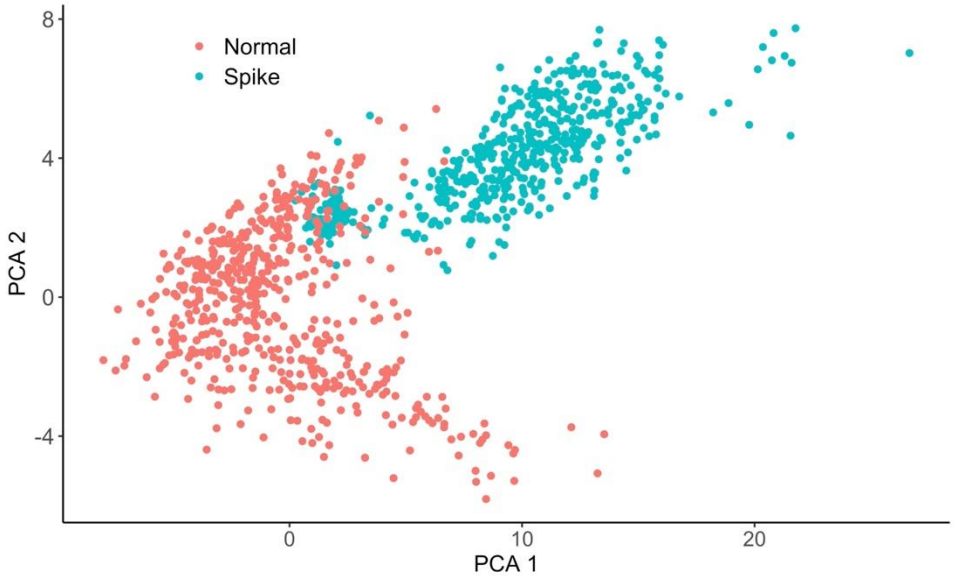


(a) Original dataset at hour 4

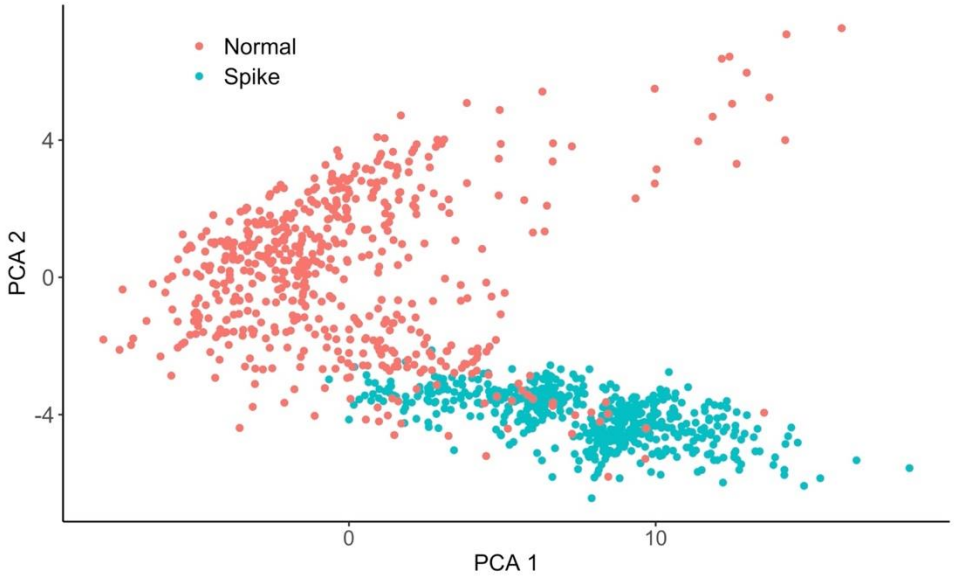


(b) Original dataset at hour 17

Figure 3.14 Scatter plot of the original dataset at hour 4 and hour 17.



(a) Oversampled dataset at hour 4



(b) Oversampled dataset at hour 17

Figure 3.15 Scatter plot of the oversampled dataset at hour 4 and hour 17

To evaluate the impact of oversampling on the price spike classification, we compared the classification results derived from the classifier that is trained by the original dataset and/or the oversampled dataset. We choose the support vector machines (SVM) method as the classifier because 1) the SVM method is able to provide an acceptable classification to unseen data as it minimizes the structural risk instead of the empirical risk [25]; 2) the SVM method enables to learn the non-linear relationship between the predictors and target variable; and 3) the classification performance of the SVM method is well proved in many publications [15]-[16]. In order to determine whether or not the electricity price is a price spike, we utilize the electricity prices of previous day as predictors. For example, to determine if the electricity price at hour 1 on August 1, 2006 is a price spike or not, we utilize the electricity prices from hour 1 to hour 24 on July 31, 2006 as predictors. Table 3.5 shows the confusion matrix derived from the SVM that is trained by the original dataset. The training period is from January 2, 2005 to July 31, 2006 and the test period is from August 1, 2006 to December 30, 2006. Within the training period, there are total 13,824 training cases and 576 cases for each hour. During the test/forecast period, there are 3,648 cases and 152 cases for each hour. From Table 3.5, the classification precision is $38/(38+47) = 44.71\%$ and recall is $38/(38+32) = 54.29\%$. Thus, the F-2 score is 0.52. Similarly, Table 3.6 shows the confusion matrix derived from the SVM that is trained by the oversampled dataset. Within the training period of January 2, 2005 to July 31, 2006, there are total 26,528 training cases and around 1,100 cases for each hour. During the test/forecast period of August 1, 2006 to December 30, 2006, there are 3,648 cases and 152 cases for each hour. From Table 3.6, the classification precision is $65/(65+42) = 60.75\%$ and recall is $65/(65+5) = 92.85\%$. Its F-2 score is 0.84.

Table 3.5 Confusion matrix derived from the SVM trained by the original dataset

Items	True Spike	True Normal
Spike Prediction	38	47
Normal Prediction	32	3,531

Table 3.6 Confusion matrix derived from the SVM trained by the oversampled dataset

Items	True Spike	True Normal
Spike Prediction	65	42
Normal Prediction	5	3,536

Table 3.7 summarizes the classification results derived from the SVM trained by the original dataset and/or the oversampled dataset. Since we care more about recall than precision as it is vital to classify as many true price spikes as possible, the recall of the SVM trained by the oversampled dataset is 85.71% that is almost 58% higher than the recall of the SVM trained by the original dataset. For the F2-score, the closer the score to 1, the better classification performance is. From Table 3.7, we can tell that the F2-score of the SVM trained by the oversampled data is 0.79 that is much larger than the F2-score of the SVM trained by the original dataset. In other words, the performance of price spike classification is significantly improved by using our proposed oversampling methods.

Table 3.7 Classification results comparation

	Original data	Oversampled data	Improvement
Precision	44.71%	60.75%	35.88%
Recall	54.29%	92.85%	71.03%
F2-score	0.52	0.84	61.31%

3.4.2 Price spike regression results

In this subsection, we focus on forecasting the numeric values of the price spikes. The top three highest day-ahead electricity prices occurred on August 1-3 are shown in Figure 3.16. As

the prices on August 2 and 3 are both much higher than the prices on August 1, therefore, we utilize those two days as examples to show the performance of our price spike numeric forecast. Using August 2, we will explain the details of the numeric value forecast of price spikes. August 3 is used to compare our price spike forecast results with the results obtained from other popular regression methods.

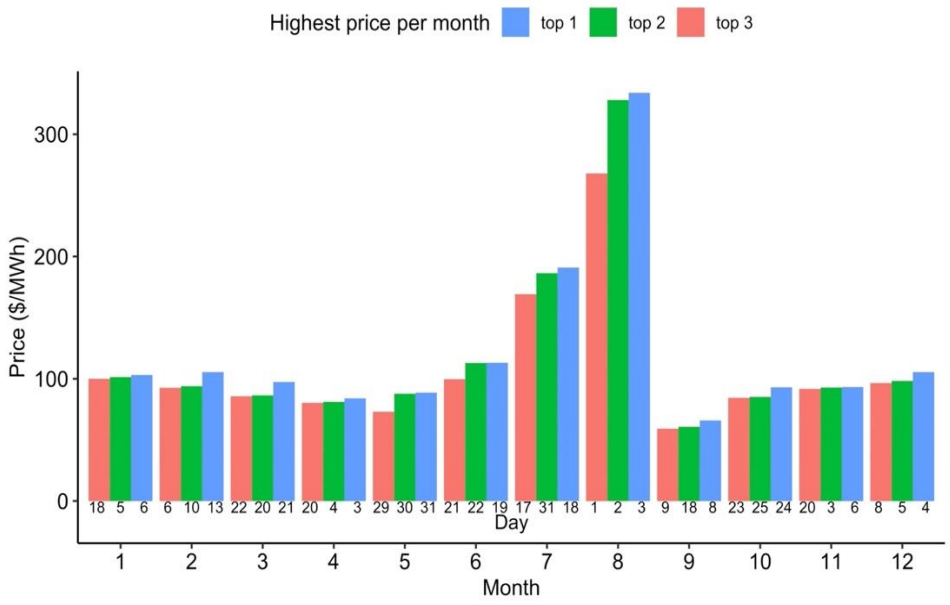


Figure 3.16 Three highest prices per month in 2006.

According to the price spike classification forecast results discussed in the previous subsection, we have identified price spikes on August 2 (e.g., hours 1-3 and 10-24). Now, we should give numeric target value to the oversampled price spike cases by using the SMOTE for regression method. As different hours have different price spike thresholds, we do need to oversample dataset for each hour. For each hour, in order to assign a numeric value to an oversampled case, we need to find several real cases that are near to the selected oversampled case and utilize the target values from the real cases to calculate the target value of the selected

oversampled case according to the formulation (2.40). For example, in order to derive the target value of a price spike case at hour 1, firstly, we measure the predictors distance between the oversampled case with 29 real price spike cases. Then, we utilize the targets of the two nearest real cases to calculate the target value of the oversampled case.

Table 3.8 Price spike forecast results on august 2, 2006

Hour	Actual Price ($/MWh)	Forecast Price ($/MWh)	Relative Error
1	99.85	100.21	0.36%
2	80.00	82.48	3.10%
3	63.55	68.47	7.73%
10	143.33	157.69	10.02%
11	169.11	176.99	4.66%
12	208.12	208.86	0.35%
13	235.87	231.26	1.95%
14	271.30	264.38	2.55%
15	301.82	293.76	2.67%
16	321.45	313.06	2.61%
17	328.05	319.03	2.75%
18	298.60	287.65	3.67%
19	265.06	251.31	5.18%
20	217.38	202.60	6.80%
21	208.92	196.84	5.78%
22	183.45	178.46	2.71%
23	136.79	142.87	4.45%
24	94.12	112.60	19.64%

After deriving the oversampled dataset for each hour, the regression algorithm trained by the oversampled dataset is used to forecast the price values at the spike hours determined by the classifier. To show the performance of the price spike forecast, we use August 2, 2006 as an example and list the numeric forecast results in Table 3.8. The highest day-ahead electricity price on August 2, 2006 occurred at hour 17 and the price was 328.05 $/MWh. Compared with the forecasted spike price in [12] that the forecast value is 289.83 $/MWh at hour 17 and the relative

error is 11.65%, our forecasted price value is 319.03$/MWh and the relative forecast error is only 2.75%. The relative error reduction is 76.39% by using our proposed oversampling methods.

To obtain final price forecast results, we place the normal forecast values at the place the classifier is regarded as normal price and place the price spike forecast values at the place that the classifier is regarded as price spike. We utilize August 3, 2006 as an example to show the final forecast results. The highest day-ahead electricity price in 2006 is 333.91$/MWh that happened at hour 17 on August 3. The AMAPE of the on-peak hour price forecast is 6.98% and the off-peak hour price forecast is 5.70%. The AMAPE over 24 hours on August 3 is 6.50%. Besides, the forecasted spike value at hour 17 is 346.98$/MWh with the relative error of 3.91%.

To further compare the forecast performances of our proposed method with other forecast methods such as neural network (NN) [10], extreme learning machine (ELM) [5], and autoregressive integrated moving average (ARIMA) [11], we utilize some off-the-shelf packages, such as ‘forecastHybrid’, ‘nnfor’, and ‘forecast’, to calculate the day-ahead electricity price on August 3, 2006. The parameters of NN, ELM and ARIMA are determined by the automatic tune process provided by those forecast packages. The forecast results of different methods are shown in Figure 3.17.

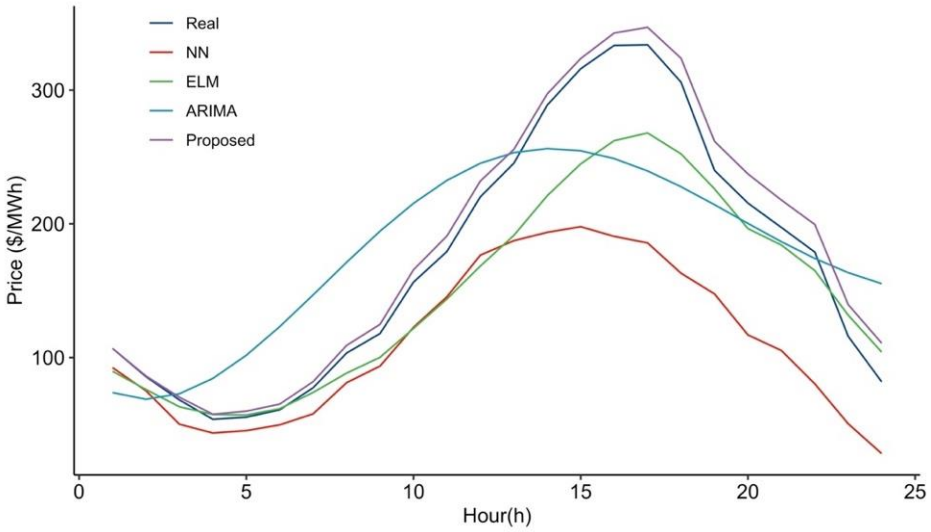


Figure 3.17 Real and forecasted day-ahead PJM electricity price (August 3, 2006).

As the ELM method can learn the complex nonlinear relationships between input and output, compared with the ARIMA method that can only find linear relationships, the forecast performance of the ELM is better than the ARIMA. The AMAPE values of the ELM and ARIMA methods are 16.38% and 26.12%, respectively. Neural network can also learn non-linear patterns between input and output. However, as the majority of electricity prices are normal prices, in order to ensure normal price forecast accuracy, the NN method barely provides accurate predictions during on-peak hours. Therefore, the AMAPE of the NN method during on-peak hours is 45.05% and during the off-peak hours is 18.82%. However, the electricity price predictions provided by our proposed method are close to the real prices during both on-peak and off-peak hours which shows that our forecast method has a better forecast performance compared with all other three traditional methods.

3.5 Case study 3: various electricity markets forecast results

To further justify the effectiveness of our proposed price forecast framework, in this section, multiple electricity markets are studied, such as MISO, ISO-NE, NYISO, SPP, and the electricity market of Spain. The selected price nodes and test dates are listed in Table 3.9. This challenging case study will predict the day-ahead electricity prices in the first seven days of each season in the year 2020: Mar 1-7 (Spring), Jun 1-7 (Summer), Sep 1-7 (Fall), and Dec 1-7, 2020 (Winter). To show the performance of our proposed short-term price forecast, the AMAPE and MAPE of forecast errors are listed in Tables 3.10 and 3.11, respectively. From Table 3.10, we can see that for each market, our proposed forecast performances for the selected weeks are consistent and acceptable. In addition, we list the MAPE values in Table 3.11. Basically, the values are acceptable, except for the SPP market. The MAPE values of the SPP market are abnormal compared with other markets. that the MAPE value of the week of March 1-7 is 263%.

Table 3.9 Selected price nodes and test dates for multiple electricity markets

Market	Node	Testing Dates
MISO	MISO System	Mar 1-7, Jun 1-7, Sep 1-7, and Dec 1-7, 2020
ISO-NE	Internal hub	Mar 1-7, Jun 1-7, Sep 1-7, and Dec 1-7, 2020
NYISO	Zone New York City	Mar 1-7, Jun 1-7, Sep 1-7, and Dec 1-7, 2020
SPP	Sppnorth	Mar 1-7, Jun 1-7, Sep 1-7, and Dec 1-7, 2020
Spain	Mainland Spain	Mar 1-7, Jun 1-7, Sep 1-7, and Dec 1-7, 2020

Table 3.10 AMAPE of multiple electricity markets

Week of 2020	MISO	ISO-NE	NYISO	SPP	Spain
Mar. 1-7	5.06%	8.46%	4.40%	11.99%	6.62%
Jun. 1-7	5.20%	5.96%	4.92%	12.28%	4.35%
Sep. 1-7	3.77%	6.48%	4.02%	10.49%	3.72%
Dec. 1-7	3.56%	7.07%	3.79%	9.62%	4.54%
Average	4.40%	6.99%	4.28%	11.10%	4.81%

Table 3.11 MAPE of multiple electricity markets

Week of 2020	MISO	ISO-NE	NYISO	SPP	Spain
Mar. 1-7	5.00%	8.50%	4.43%	263.00%	6.74%
Jun. 1-7	5.37%	5.88%	4.67%	23.42%	4.45%
Sep. 1-7	3.90%	6.43%	3.95%	25.77%	3.72%
Dec. 1-7	3.48%	7.19%	3.77%	10.34%	4.62%
Average	4.44%	7.00%	4.20%	80.64%	4.88%

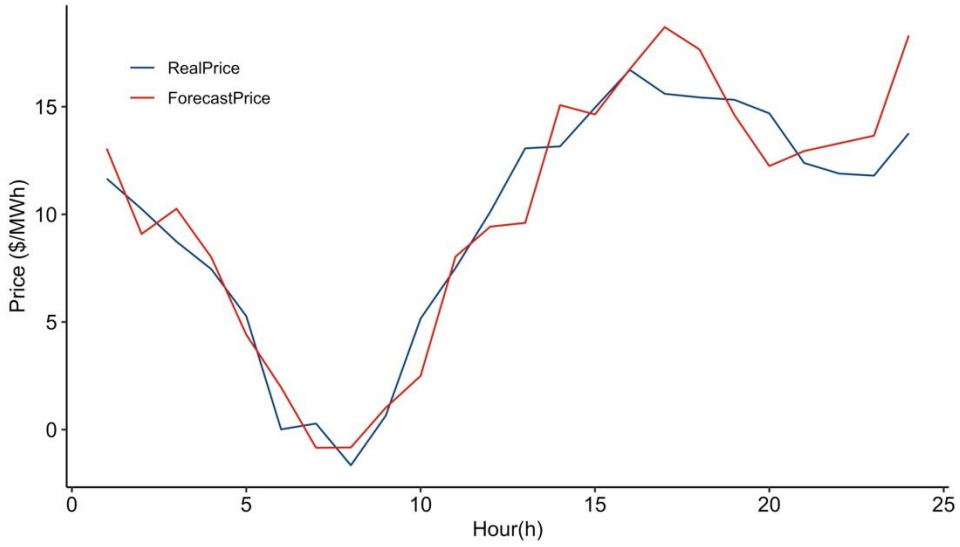


Figure 3.18 Real and forecasted day-ahead SPP electricity price (March 2, 2020).

To understand more about the performance of our forecast methods on the SPP market, we also demonstrate the price forecast results on March 1 and September 2, 2020 in Figure 3.19 and 3.20, respectively. Obviously, the forecast results are close to the real electricity prices of the SPP market. As the lowest price on March 1 is 6.33$/MWh and on September 2 is 14.86$/MWh (none of the real prices in those two days are close to zero), the MAPE value on March 1 is 8.08% and on September 2 is 5.80%, which is reasonable (note that the AMAPE values on March 1 and September 2 are 7.02% and 5.33%, respectively). Besides, while March 1 and 2 are

consecutive days, the price patterns that are shown in Figure 3.18 and 3.19 are different. For example, the peak price on March 2 is at hour 16 but the peak price is at hour 1 on March 1. Although the electricity price patterns of those two days are different, from Figure 3.18 and 3.19, we can tell that our price forecasts are close to the real prices. Again, the forecast performance shows that our forecast method is capable to handle different markets under different situations

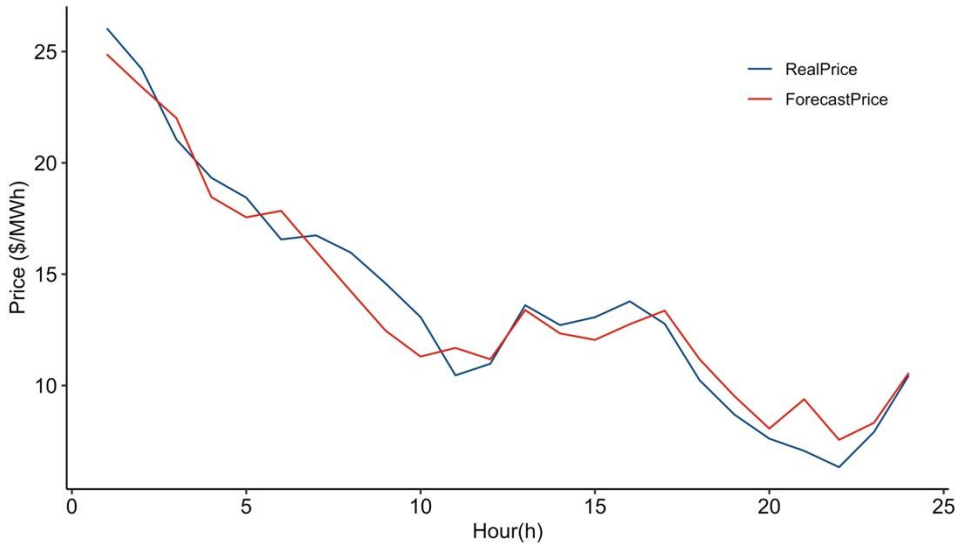


Figure 3.19 Real and forecasted day-ahead SPP electricity price (March 1, 2020).

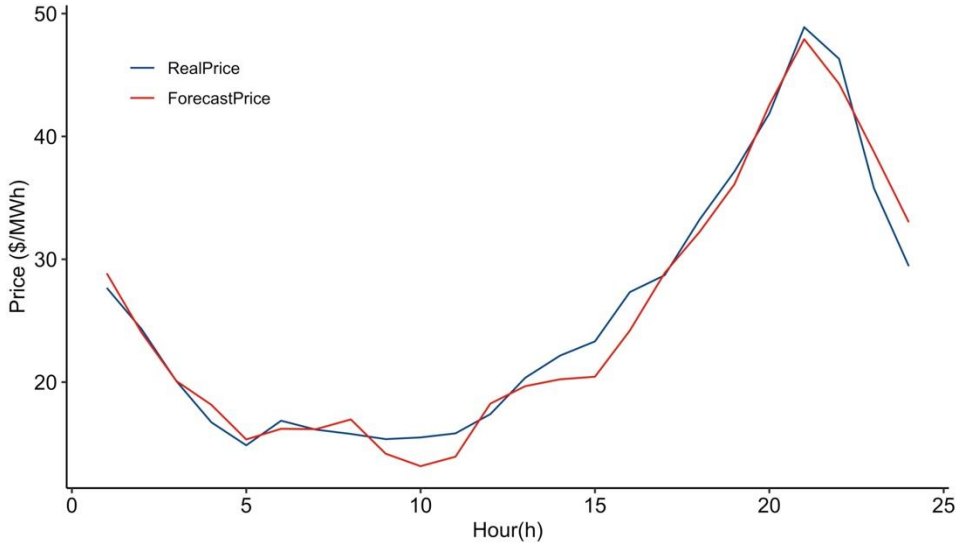


Figure 3.20 Real and forecasted day-ahead SPP electricity price (September 2, 2020)

To provide more detailed assessments, we also use the relative mean absolute error (rMAE) as a metric to measure the point forecasts accuracy as the authors [60] argue that the scaled error metric facilitates comparison between datasets. Besides, the Giacomini-White (GW) test is adopted to determine whether the predictions derived from the proposed method are statistically accurate than the forecasts derived from the benchmark models. The open-access benchmark models [60] we adopt are the Lasso Estimated Auto Regression (LEAR) and the Deep Neural Network (DNN) where the LEAR is a linear statical model and the DNN is able to model the nonlinear relationships between the predictors and the target variables.

As shown in Table 3.12, the rMAP value of the proposed method are always lower than the benchmark methods for every market in the selected seasons. Besides, to determine whether the difference of forecast accuracy is statistically significant, the p-values of the GW test are listed in Table 3.13. As the p-values close to 0 means there are significant differences between

forecast results, combined with the accuracy results shown in Table 3.12, we can tell that our price forecasts are statistically better than the predictions derived from the benchmark models.

Table 3.12 rMAP of multiple electricity markets

Date in 2020	**MISO**			**ISO-NE**			**NYISO**			**Spain**		
	LEAR	DNN	Prop.	LEAR	DNN	Prop.	LEAR	DNN	Prop.	LEAR	DNN	Prop.
Mar.1-7	0.52	0.62	0.29	0.85	0.62	0.45	0.82	1.04	0.49	0.88	0.92	0.55
Jun.1-7	0.73	0.73	0.28	1.08	0.73	0.47	0.64	0.75	0.29	0.74	0.94	0.34
Sep.1-7	0.89	0.75	0.28	0.69	0.75	0.38	0.64	0.78	0.27	0.59	0.59	0.49
Dec.1-7	0.95	0.91	0.37	0.83	0.91	0.42	0.7	1	0.33	0.77	0.75	0.53
Average	0.77	0.75	0.31	0.86	0.75	0.43	0.7	0.89	0.34	0.74	0.8	0.48

Table 3.13 MAPE of multiple electricity markets

Comparison	**MISO**	**ISO-NE**	**NYISO**	**Spain**
LEAR/Prop.	3.89e-07	2.41e-06	9.57e-06	7.65e-04
DNN/Prop.	2.75e-06	5.27e-06	4.80e-07	9.36e-05

3.6 Conclusion

In chapter, numerous case studies for different electricity markets, such as PJM, MISO, ISO-NE, NYISO, et cetera, were discussed in detail and proper evaluation matrices were utilized to assess the overall performance of our proposed price forecast framework. According to the analysis on the used methodologies and numerical comparison studies, our proposed work is capable to predict short-term electricity prices of different markets with acceptable accuracy and reliable performance.

CHAPTER IV

PROBABILISTIC FORECAST AND RESULTS

In this chapter, an optimal prediction interval forecast method is presented to capture the uncertainties within electricity prices. Section 4.1 presents the metrics that are used to measure the performance of prediction interval forecasts. The formulations of the proposed optimal prediction interval problem and the solving method are shown in section 4.2. Besides, to facilitate understanding, we list the flowchart and detailed calculation procedures at the end of section 4.2. To justify the effectiveness of the proposed methods, case study results are shown in section 4.3. The conclusions are drawn in section 4.4.

4.1 Prediction interval metrics

In order to estimate the performance of prediction interval forecasts, we introduce some common measurements in this chapter that one is reliability, and the other one is sharpness [75]. The detailed mathematic formations of the two metrics are shown below.

4.1.1 Reliability

Reliability is one of the key measurements to evaluate whether the forecast prediction interval aligns with the pre-defined confidence level. To provide detailed explanations, we introduce two terminologies that one is the prediction interval nominal confidence (PINC) and the other is the prediction interval coverage probability (PICP) [36]-[37]. The PINC indicates the pre-defined confidence level, and the formulation is shown below

$$PINC = 100 \times (1 - \beta)\% \tag{4.1}$$

, where β is the pre-defined probability of error [38]. The PICP indicates the percentage of forecasts within the predefined interval. The mathematic expression is shown in equation (4.2)

$$PICP = 100 \times \frac{1}{N}\sum_{i=1}^{N} C_i \tag{4.2}$$

where the N is the number of cases in the test dataset. The C_i implies whether the forecast target value is within the predefined interval or not.

$$C_i = \begin{cases} 1 & t_i \in \left[q_i^{\underline{\alpha}}, q_i^{\overline{\alpha}}\right] \\ 0 & t_i \notin \left[q_i^{\underline{\alpha}}, q_i^{\overline{\alpha}}\right] \end{cases} \tag{4.3}$$

The t_i is the predicted target value. The $\underline{\alpha}$ and $\overline{\alpha}$ are the upper and lower quantile proportions and the $q_i^{\underline{\alpha}}$ and $q_i^{\overline{\alpha}}$ represent the upper and lower quantile of the prediction interval, respectively [39].

In order to evaluate whether the experimental prediction interval aligns with the pre-defined one, we measure the deviation of PICP from PINC, and called the averaged difference between PICP and PINC as the average coverage error (ACE).

$$ACE = PICP - PINC \tag{4.4}$$

The value of ACE could be positive or negative that indicates the PICP value is higher or lower than the PINC. The closer ACE to zero, the better prediction interval forecasts we have.

4.1.2 Sharpness

A large positive ACE value means the electricity price uncertainties have been amplified. The consequence of using the over-width prediction interval is that market participants might make a conservative decision and result in benefits reduction. Therefore, we also need to

measure the sharpness of the prediction interval forecasts. The sharpness means the width of the prediction interval [40] and is calculated by using equation (4.5).

$$PIAW = \frac{1}{N}\sum_{i=1}^{N}\left(q_i^{\overline{\alpha}} - q_i^{\alpha}\right) \tag{4.5}$$

The PIAW is short for prediction interval average width.

When using the sharpness to evaluate the prediction interval forecasts, the closer the PIAW value reaches zero the sharper prediction interval we will get. Then we expect that the calculated PIAW as small as possible. However, if our probabilistic forecasts reach the lowest PIAW value, this means the upper and lower bound of the prediction interval are the same and our probabilistic forecasts become point forecasts. In that situation, we cannot quantify the uncertainty within the future electricity prices. Therefore, we need to ensure reliability and try to minimize the width between the upper and lower bound of the prediction interval forecasts.

4.2 Proposed optimal prediction interval method

In this section, the optimal prediction interval problem is formulated in subsection 4.2.1. In subsection 4.2.2, we introduce the extreme learning machine (ELM) to provide the upper and lower bounds of prediction interval. In subsection 4.2.3, an augmented Lagrangian method is adopted to solve the optimal prediction interval problem. The detailed solution steps and flowchart are shown in the last subsection.

4.2.1 Formulations of the optimal prediction interval problem

Before introducing the formulations, first, we need to define the meaning of optimal prediction interval. In this chapter, the optimal prediction interval is the prediction interval that takes both reliability and sharpness into consideration. To fulfill the reliability requirement, the objective function is shown below

$$min \sum_{i=1}^{N} \left[\rho_{\overline{\alpha}}(y_i - q_i^{\overline{\alpha}}) + \rho_{\underline{\alpha}}(y_i - q_i^{\underline{\alpha}}) \right] \tag{4.6}$$

, where y is the observed target value and i is the index of cases. The ρ represents a function with a tiled form that is called tick or pinball loss function [41] [73]. The formulation of the pinball loss function [77][79][80] is listed in equation (4.7)

$$\rho = \begin{cases} \alpha u & if\ u \geq 0 \\ (\alpha - 1)u & if\ u < 0 \end{cases} \tag{4.7}$$

, where the u represents the difference between the observed target value and the corresponding upper or lower predicted quantiles. The pinball loss function can be visualized in Figure 4.1.

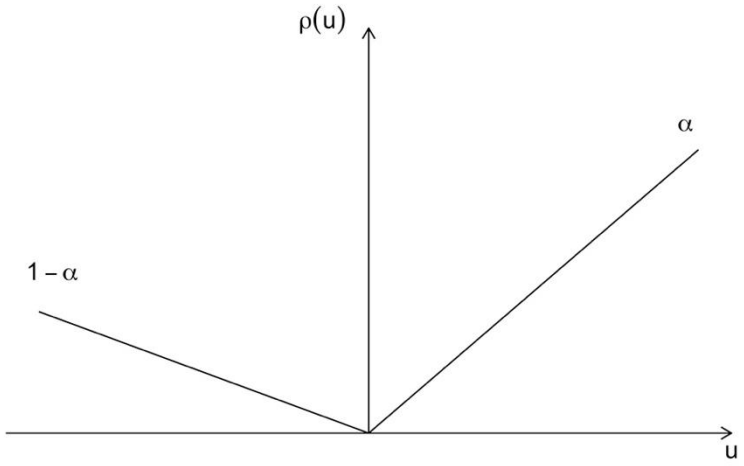


Figure 4.1 Pinball loss function

Besides, to ensure we have a sharper prediction interval, we also place the differences between the upper and lower prediction interval bounds into our objective function. The sharpness requirement is shown in formulation (4.8).

$$min \sum_{i=1}^{N} \left(q_i^{\overline{\alpha}} - q_i^{\underline{\alpha}} \right) \tag{4.8}$$

After showing the objective function, we introduce several constraints that related to the optimal prediction interval problem. In order to avoid quantile crossing situation that is the lower quantile forecasts larger than the upper quantile predictions, we utilize the non-crossing constraint (4.9) to ensure the quantile forecast monotonically increase as the quantile probability increase [42].

$$q_i^{\underline{\alpha}} \leq q_i^{\overline{\alpha}} \qquad i = 1 \dots N \tag{4.9}$$

To ensure the width of the predictions aligns with the pre-defined confidence level, the next set of constraints are related to the width of the prediction intervals. Researchers that use parametric ways to calculate the prediction intervals normally assume that the electricity price errors follow a normal distribution that has a symmetric spread around the mean values [7], [43]. Therefore, the central prediction interval (CPI) is a common way to define the bounds of prediction intervals. The upper and lower bounds are calculated by using the point forecasts to add or subtract the critical value of the normal distribution multiplying the square root of the variance, respectively. However, [39] points out if the target value distribution is skewed, the CPI might be a conservative way to derive the prediction interval width. Therefore, in our study, we utilize the probability mass bias (PMB) metric [45] to measure the deviation of the proposed prediction interval from the central prediction interval

$$PMB = (1 - \overline{\alpha}) - \underline{\alpha} \; = \; -2\delta \tag{4.10}$$

, where the δ indicates the deviation of the upper or lower bound of the proposed prediction interval from the corresponding central prediction interval bounds. If we want to make sure that the confidence interval is $(1 - \beta)\%$, we can use the following constraints.

$$\overline{\alpha} = 1 - \frac{\beta}{2} + \delta \tag{4.11}$$

$$\underline{\alpha} = \frac{\beta}{2} + \delta \tag{4.12}$$

And to guarantee the range of $\overline{\alpha}$ and $\underline{\alpha}$ within interval [0,1], we include constraint (4.13).

$$-\frac{\beta}{2} \leq \delta \leq \frac{\beta}{2} \tag{4.13}$$

From Fig. 4.1, we can tell that the pinball loss function is not differentiable at the origin. In order to facilitate calculation, according to [45], we can linearize the pinball loss objective function (4.6) using the following equations and constraints.

$$min \sum_{i=1}^{N} \overline{\alpha}\xi_{i,+}^{\overline{\alpha}} + (1-\overline{\alpha})\xi_{i,-}^{\overline{\alpha}} + \underline{\alpha}\xi_{i,+}^{\underline{\alpha}} + \left(1-\underline{\alpha}\right)\xi_{i,-}^{\underline{\alpha}} \tag{4.14}$$

$$y_i - q_i^{\overline{\alpha}} = \xi_{i,+}^{\overline{\alpha}} - \xi_{i,-}^{\overline{\alpha}} \; i = [1 \dots N] \tag{4.15}$$

$$y_i - q_i^{\underline{\alpha}} = \xi_{i,+}^{\underline{\alpha}} - \xi_{i,-}^{\underline{\alpha}} \; i = [1 \dots N] \tag{4.16}$$

$$\xi_{i,+}^{\overline{\alpha}}, \xi_{i,-}^{\overline{\alpha}}, \; \xi_{i,+}^{\underline{\alpha}}, \xi_{i,-}^{\underline{\alpha}} \geq 0 \;\; i = [1 \dots N] \tag{4.17}$$

In order to consider both reliability and sharpness, we combine (4.14) and (4.8) to get a new objective formulation

$$\begin{aligned} min \sum_{i=1}^{N} (w\big[\overline{\alpha}\xi_{i,+}^{\overline{\alpha}} + (1-\overline{\alpha})\xi_{i,-}^{\overline{\alpha}} + \underline{\alpha}\xi_{i,+}^{\underline{\alpha}} + \left(1-\underline{\alpha}\right)\xi_{i,-}^{\underline{\alpha}}\big] \\ +\big[q_i^{\overline{\alpha}} - q_i^{\underline{\alpha}}\big]) \end{aligned} \tag{4.18}$$

, where the w is a weight factor that indicates the importance of the reliability over sharpness.

The overall optimal prediction interval problem is formulated below.

$$s.t. \begin{cases} Obj.(4.18) \\ Non-crossing\ constraints: (4.9) \\ Prediction\ interval\ constraints: (4.11)-(4.13) \\ Pinball\ loss\ linearization: (4.15)-(4.17) \end{cases} \tag{4.19}$$

4.2.2 Extreme learning machine

The extreme learning machine (ELM) is a type of single-hidden layer feed-forward neural network (SLFN) that is proposed by Huang et al. [46]. Compared with other machine learning algorithms such as the least square support vector machine (LS-SVM), the ELM could achieve a better regression performance. Besides, in theory, ELM could estimate any continuous target function [47]. One of the salient characteristics of the method is that the computational burden of the ELM method is low. Compared with neural networks that require multiple back-and-forth feed-forward and backpropagation calculations, the input weights and the hidden layer biases of an ELM are randomly initialized and without further tuning. And the output weights are calculated by using a matrix multiplication method. Based on the characteristics of the ELM, we choose this method to forecast the upper and lower bound of the optimal prediction intervals.

We utilize $(\boldsymbol{x}_i, y_i), i = 1, \dots, N$ to indicate the training set with the number of N cases. $\boldsymbol{x}_i$ is the predictor vector where $\boldsymbol{x} \in \boldsymbol{R}^n$ and y_i is the observed target value of the case i. The symbol $t_i$is used to denote the predicted target value. The hidden node activation function is denoted using the symbol g. If we adopt an ELM with a number of M hidden nodes to estimate y_i from $\boldsymbol{x}_i$, we can use the following formulation to show the relationship between the predictors and target values.

$$\sum_{j=1}^{M} o_j g\left(\boldsymbol{\omega}_j \boldsymbol{x}_i + b_j\right) = t_i \; i = 1, \dots, N \tag{4.20}$$

The symbol $\boldsymbol{\omega}_j$ is the input weight vector that connects the input node with the hidden node j and b_j is the bias of node j. Besides, o_j means the output weight vector of the hidden node j that connects with the output node.

The equation (4.20) can also be presented using a matrix form to facilitate expression.

$$\boldsymbol{Ho} = \boldsymbol{t} \tag{4.21}$$

The $\boldsymbol{H}$ is the hidden layer output matrix. The expended $\boldsymbol{H}$ is shown in (4.22)

$$\boldsymbol{H} = \begin{bmatrix} g(\boldsymbol{\omega}_1 \boldsymbol{x}_1 + b_1) & \cdots & g(\boldsymbol{\omega}_M \boldsymbol{x}_1 + b_M) \\ \vdots & \ddots & \vdots \\ g(\boldsymbol{\omega}_1 \boldsymbol{x}_N + b_1) & \cdots & g(\boldsymbol{\omega}_M \boldsymbol{x}_N + b_M) \end{bmatrix}_{N \times M} \tag{4.22}$$

The output weight vector $\boldsymbol{o}$ and the predicted target vector $\boldsymbol{t}$ are shown below.

$$\boldsymbol{o} = \begin{bmatrix} o_1 \\ \vdots \\ o_M \end{bmatrix}_{M \times 1} \quad and \; \boldsymbol{t} = \begin{bmatrix} t_1 \\ \vdots \\ t_N \end{bmatrix}_{N \times 1} \tag{4.23}$$

The minimization problem (4.24) is use to make sure the ELM forecast values are close to the observed target values.

$$min_{\boldsymbol{\beta}} \|\boldsymbol{Ho} - \boldsymbol{y}\|_2^2 \tag{4.24}$$

According to [46], since the input weights $\boldsymbol{\omega}_j$ and the hidden layer biases b_j do not need to be modified after initialization, we can treat the hidden layer output matrix $\boldsymbol{H}$ as a constant. Therefore, in the minimization problem (4.24), we only need to tune the output weight vector $\boldsymbol{o}$. The (4.24) is actually a least square problem and the optimal $\boldsymbol{o}$ can be derived analytically.

$$\hat{\boldsymbol{o}} = \boldsymbol{H}^{\dagger} \boldsymbol{y} \tag{4.25}$$

The $\boldsymbol{H}^{\dagger}$ is the Moore–Penrose generalized inverse of the hidden layer output matrix. The generalized inversion is used here to relief calculation difficulties if $\boldsymbol{H}$ is a singular matrix or do not have a squared form.

Compared with neural networks that use numbers of iterations to tune parameters, the output weights of the ELM can be derived by calculating a generalized matrix inversion and then use the inversed matrix to multiply the observed target vector. Besides, other parameters are fixed at the beginning. Therefore, the training speed of the ELM method is quick. Besides, we

can choose different activation functions to model the nonlinear relationships between the predictors and target values.

4.2.3 Reformed formulations of the optimal prediction interval problem

In this subsection, we will present a modified optimal prediction interval problem based on the optimal problem formulated in 4.2.1 and the forecast method introduced in 4.2.2.

Based on the characteristic of the ELM, only the output weight parameters need to be tuned. Therefore, by using the ELM to predict the upper and lower bound of prediction interval, the objective (4.18) can be rewritten.

$$\min_{\boldsymbol{x}} \sum_{i=1}^{N} (w[\,\overline{\alpha}\xi_{i,+}^{\overline{\alpha}} + (1-\overline{\alpha})\xi_{i,-}^{\overline{\alpha}} + \underline{\alpha}\xi_{i,+}^{\underline{\alpha}} + (1-\underline{\alpha})\xi_{i,-}^{\underline{\alpha}}] + [\boldsymbol{h}_i^T \boldsymbol{o}_{\overline{\alpha}} - \boldsymbol{h}_i^T \boldsymbol{o}_{\underline{\alpha}}]) \tag{4.26}$$

$\boldsymbol{x} = \{\overline{\alpha}, \underline{\alpha}, \delta, \boldsymbol{o}_{\overline{\alpha}}, \boldsymbol{o}_{\underline{\alpha}} \xi_{i,+}^{\overline{\alpha}}, \xi_{i,-}^{\overline{\alpha}}, \xi_{i,+}^{\underline{\alpha}} \xi_{i,-}^{\underline{\alpha}}\}$. The $\boldsymbol{h}_i$ is the hidden layer output vector for the case i and the superscript T means transpose operation. Besides, the $\boldsymbol{o}_{\overline{\alpha}}$ and $\boldsymbol{o}_{\underline{\alpha}}$ are the output weight vector of the ELM for the upper and lower quantile of the prediction interval, respectively.

Except the objective function, we also modified several sets of constraints. As the non-crossing constraint (4.9) is related to the upper and lower quantile of the prediction interval, we changed it into the following form.

$$\boldsymbol{h}_i^T \boldsymbol{o}_{\underline{\alpha}} \leq \boldsymbol{h}_i^T \boldsymbol{o}_{\overline{\alpha}} \quad i = 1 \ldots N \tag{4.27}$$

Since the prediction interval constraints are not related to quantile prediction, we leave them unchanged. For the constraints related to the objective function linearization, we substitute (4.15) and (4.16) by using equation (4.28) and (4.29) as constraint (4.15) and (4.16) are related to the quantile forecast.

$$y_i - \boldsymbol{h}_i^T \boldsymbol{o}_{\overline{\alpha}} = \xi_{i,+}^{\overline{\alpha}} - \xi_{i,-}^{\overline{\alpha}} \; i = [1 \ldots N] \tag{4.28}$$

$$y_i - \boldsymbol{h}_i^T \boldsymbol{o}_{\underline{\alpha}} = \xi_{i,+}^{\underline{\alpha}} - \xi_{i,-}^{\underline{\alpha}} \; i = [1 \dots N] \tag{4.29}$$

After the adjustment, the reformed optimal prediction interval problem is shown in (4.30)

$$s.t. \begin{cases} Obj.(4.26) \\ Non-crossing\ constraints: (4.27) \\ Prediction\ interval\ constraints: (4.11) - (4.13) \\ Pinball\ loss\ linearization: (4.17)\ and\ (4.28) - (4.29) \end{cases} \tag{4.30}$$

4.2.4 Augmented Lagrangian method

The augmented Lagrangian method is used to solve the reformed optimization problem in this subsection. As the nonlinear terms, such as $\overline{\alpha}\xi_{i,+}^{\overline{\alpha}}$, exist in the objective function (4.26), and there are several equality and inequality constraints in the constraint set, we can tell that the optimization problem (4.30) is a nonlinear optimization problem with different types of constraints. Therefore, the off-the-shelf solvers are hard to directly apply on the problem. However, as those solvers are fast and reliable, in this paper, an Augmented Lagrangian method [48] is used to modify the optimization problem so that solvers only need to calculate an unconstrained optimization problem in each iteration.

To avoid complicated expression and facilitate generalization, we simplify the reformed optimal prediction interval problem (4.30). The objective function in (4.30) is denoted as $f(\boldsymbol{x})$. The constraints are categorized into two classes that one set of constraints is equality constraints and the other set is inequality constraints

$$P = \begin{matrix} \{c_k(\boldsymbol{x}) = 0 & k = 1 \dots r \\ c_k(\boldsymbol{x}) \geq 0 & k = r+1 \dots m\} \end{matrix} \tag{4.31}$$

We utilize k as the constraints index. Symbol r and $m - r$ are the number of equality and inequality constraints, respectively.

By using the generalized objective function and constraints, we change the reformed optimization problem with constraints into an unconstrained optimization problem by using the augmented Lagrangian method.

$$\varphi(\boldsymbol{x}, \boldsymbol{\lambda}, \sigma) = f(\boldsymbol{x}) - \boldsymbol{\lambda}^T \boldsymbol{d}(\boldsymbol{x}) + \frac{1}{2}\sigma \boldsymbol{d}(\boldsymbol{x})^T \boldsymbol{d}(\boldsymbol{x}) \tag{4.32}$$

The elements of vector $\boldsymbol{\lambda}$ are the Lagrangian multipliers for constraints and σ is a positive number that penalizes active constraints violation. The $\boldsymbol{d}(\boldsymbol{x})$ is defined as

$$d_k(\boldsymbol{x}) = \begin{cases} c_k(\boldsymbol{x}) & k < r \text{ or } c_k(\boldsymbol{x}) \leq \delta \\ 0 & otherwise \end{cases} k = 1 \dots m \tag{4.33}$$

, where δ is a small positive number that is use to define the active inequality constraints. The meaning of $\boldsymbol{d}(\boldsymbol{x})$ is to ensure the objective with augmented Lagrangian form only penalizes active constraints.

According to [49], Fletcher shows that if the σ is larger than a finite number $\hat{\sigma}$ and $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$, then $\boldsymbol{x}^*$ is an unconstrained local minimizer of $\varphi(\boldsymbol{x}, \boldsymbol{\lambda}^*, \sigma)$.

$$\boldsymbol{x}^*{}_{\boldsymbol{\lambda},\sigma} = \underset{\boldsymbol{x}}{argmin}\, \varphi(\boldsymbol{x}, \boldsymbol{\lambda}, \sigma) \tag{4.34}$$

Therefore, if $\sigma > \hat{\sigma}$,

$$\boldsymbol{x}^*{}_{\boldsymbol{\lambda},\sigma} = \boldsymbol{x}^* \tag{4.35}$$

So, if we utilize a large enough σ and have $\boldsymbol{\lambda}^*$, the optimal solutions of the unconstrained augmented Lagrangian optimization problem equal to the optimal solutions of the constrained one. Therefore, given a large σ and iteratively updating $\boldsymbol{\lambda}$ to the $\boldsymbol{\lambda}^*$, we can get the solutions of the proposed optimal prediction interval problem from solving a set of unconstrained problems.

In order to update $\boldsymbol{\lambda}$, we let $\sigma = \sigma_{fix}$ and form a function that is only related to the Lagrangian vector $\boldsymbol{\lambda}$

$$\psi(\boldsymbol{\lambda}) = \varphi\left(\boldsymbol{x}_{\boldsymbol{\lambda}}, \boldsymbol{\lambda}, \sigma_{fix}\right) \tag{4.36}$$

, where $\boldsymbol{x}_{\boldsymbol{\lambda}}$ is a minimizer of the augmented Lagrangian function given $\boldsymbol{\lambda}$ and σ_{fix}. To find a direction $\boldsymbol{\eta}$ that is used to update $\boldsymbol{\lambda}$, by using the Taylor expansion at $\boldsymbol{\lambda}$, equation (4.36) can be expended as

$$\begin{aligned} \psi(\boldsymbol{\lambda} + \boldsymbol{\eta}) &= \psi(\boldsymbol{\lambda}) + \boldsymbol{\eta}^{\boldsymbol{T}}\psi'(\boldsymbol{\lambda}) + O(\|\boldsymbol{\eta}\|^2) \\ &= \psi(\boldsymbol{\lambda}) - \boldsymbol{\eta}^{\boldsymbol{T}} d(\boldsymbol{x}_{\boldsymbol{\lambda}}) \end{aligned} \tag{4.37}$$

As $\boldsymbol{x}_{\boldsymbol{\lambda}}$ is a minimizer of the augmented Lagrangian function (4.32), then,

$$\varphi'_{\boldsymbol{x}}\left(\boldsymbol{x}_{\boldsymbol{\lambda}}, \boldsymbol{\lambda}, \sigma_{fix}\right) = 0 \tag{4.38}$$

which means

$$f'(\boldsymbol{x}_{\boldsymbol{\lambda}}) - \boldsymbol{J}_{\boldsymbol{d}}(\boldsymbol{x}_{\boldsymbol{\lambda}})\left[\boldsymbol{\lambda} - \sigma_{fix}\boldsymbol{d}(\boldsymbol{x}_{\boldsymbol{\lambda}})\right] = 0 \tag{4.39}$$

, where $\boldsymbol{J}_{\boldsymbol{d}}$ means Jacobian matrix. Considering the KKT stationary condition of a Lagrangian function $L'_{\boldsymbol{x}}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = 0$

$$f'(\boldsymbol{x}_{\boldsymbol{\lambda}}) - \boldsymbol{J}_{\boldsymbol{d}}(\boldsymbol{x}_{\boldsymbol{\lambda}})\boldsymbol{\lambda}^* = 0 \tag{4.40}$$

Comparing (4.40) with (4.39), assuming $\boldsymbol{x}$ is at $\boldsymbol{x}^*$, then, we can get the direction $\boldsymbol{\eta}$ to update $\boldsymbol{\lambda}$.

$$\begin{aligned} \boldsymbol{\lambda}^* &= \boldsymbol{\lambda} + \boldsymbol{\eta} \\ &\approx \boldsymbol{\lambda} - \sigma_{fix}\boldsymbol{d}(\boldsymbol{x}_{\boldsymbol{\lambda}}) \end{aligned} \tag{4.41}$$

At this time, we know how to update the Lagrangian multiplier, however, we cannot in advance decide which inequality constraints are active. Therefore, by adding auxiliary variables, we reform the inequality constraints in (4.31) into equality constraints.

$$\begin{cases} c_k(\boldsymbol{x}) - s_k = 0 \\ s_k \geq 0 \end{cases} k = r+1 \ldots m \tag{4.42}$$

Then, the augmented Lagrangian function $\varphi(\boldsymbol{x}, \boldsymbol{\lambda}, \sigma)$ is change to $\varphi(\boldsymbol{x}, \boldsymbol{\lambda}, \sigma, \boldsymbol{s})$.

$$\varphi(\boldsymbol{x}, \boldsymbol{\lambda}, \sigma, \boldsymbol{s}) = f(\boldsymbol{x}) - \sum_{k=1}^{r} \lambda_k c_k(\boldsymbol{x}) + \frac{1}{2}\sigma \sum_{k=1}^{r} [c_k(\boldsymbol{x})]^2$$
$$\sum_{k=r+1}^{m} \lambda_k [c_k(\boldsymbol{x}) - s_k] + \frac{1}{2}\sigma \sum_{k=r+1}^{m} [c_k(\boldsymbol{x}) - s_k]^2 \quad (4.43)$$

Still, we have introduced some inequality constraints that are $s_k \geq 0, k = r+1 \ldots m$. Although we have a simpler form of inequality constraints, we still want to avoid the auxiliary variables and inequality constraints.

Given $\boldsymbol{\lambda}$ and σ, the minimizers of the augmented Lagrangian function $\varphi(\boldsymbol{x}, \boldsymbol{\lambda}, \sigma, \boldsymbol{s})$ are denoted as $(\boldsymbol{x}_{\boldsymbol{\lambda},\sigma}, \boldsymbol{s}_{\boldsymbol{\lambda},\sigma})$. The characteristics of the optimal auxiliary variables are that

$$s_k = 0 \; or \; \partial\varphi/\partial s_k = 0 \quad k = r+1 \ldots m \quad (4.44)$$

The $s_k = 0$ happens if the inequality constraints are inactive. On the other hand, $\partial\varphi/\partial s_k = 0$ is the situation that inequality constraint k is active. Based on formulation (4.43), the partial derivative can be written as

$$\frac{\partial \varphi}{\partial s_k} = \lambda_k - \sigma\,(c_k(\boldsymbol{x}) - s_k) = 0 \quad k = r+1 \ldots m \quad (4.45)$$

Therefore, we can derive that $s_k = c_k(\boldsymbol{x}) - \lambda_k/\sigma$ if inequality constraints are inactive. So, (4.44) can be reformed as

$$s_k = max\,\{0, c_k(\boldsymbol{x}) - \lambda_k/\sigma\} \quad k = r+1 \ldots m \quad (4.46)$$

When performing the iterative calculation, by substituting (4.46) into (4.43), we can eliminate the auxiliary variable s_k and the inequality constraints. The augmented Lagrangian function and the reformed equality constraints are shown in (4.47) and (4.48).

$$\varphi(\boldsymbol{x}, \boldsymbol{\lambda}, \sigma) = f(\boldsymbol{x}) - \boldsymbol{\lambda}^T \boldsymbol{e}(\boldsymbol{x}) + \frac{1}{2}\sigma \boldsymbol{e}(\boldsymbol{x})^T \boldsymbol{e}(\boldsymbol{x}) \quad (4.47)$$

$$e_k(\boldsymbol{x}) = \begin{cases} c_k & if\ k \leq r\ or\ c_k \leq \lambda_k/\sigma \\ \lambda_k/\sigma & if\ k > r\ and\ c_k > \lambda_k/\sigma \end{cases} \quad k = 1 \ldots m \quad (4.48)$$

At this point, we have eliminated all inequality constraints and auxiliary variables. The overall procedure to iteratively calculate the optimization problems includes three major steps. The first step is to calculate the minimizer $\boldsymbol{x}$ of the unconstrained (4.47) given $\boldsymbol{\lambda}$ and σ. Then, the second step is to update $\boldsymbol{\lambda}$ by using (4.41) or increase the value of σ to enforce constraints. The third step is to determine whether stop iteration or not based on the pre-defined stopping criteria. The detailed steps to derive the optimal prediction interval are shown in the following subsection.

4.2.5 Flowchart and major steps

In Figure 4.2, we present the flowchart to solve the proposed optimal prediction interval problem. Besides, the detailed descriptions of each major step are also listed below.

Step 1: Parameter initialization: There are two sub-steps within this major step that one is the ELM parameter initialization and the other is the augmented Lagrangian function initialization. For the ELM parameter initialization, given there are number of M hidden nodes, we randomly assign the input weights vector $\boldsymbol{\omega}$ and the hidden layer bias b for each of them. For the augmented Lagrangian function initialization, before iteratively solving the optimal problem, initially, we let the Lagrangian multiplier vector $\boldsymbol{\lambda} = \mathbf{0}$ and assign a positive number to the penalty parameter such as $\sigma = 1$. Besides, the iteration index l is set to 0 at this step.

Step 2: Problem optimization and value calculation: This step is to calculate the unconstrained optimization problem where the objective function is the augmented Lagrangian function with the given Lagrangian multipliers and penalty parameter. At the beginning of the iterative calculation, as number of loops is one of the stopping criteria, we increase and record the iteration number. As the largest constraint violation is the other stopping criteria, after

solving the unconstrained problem, we insert the derived minimizer into constraint set (4.48) and record the largest constraint violation

Step 3: Parameter updating: At the beginning of this step, we will decide whether to update Lagrangian multipliers $\boldsymbol{\lambda}$ or update the penalty parameter σ. If the maximum constraint violation V derived by the current minimizer reduces three-quarters of the previous maximum constraint violation $V \leq V_{prev}/4$, then, we update the Lagrangian multipliers by using formulation (4.41). And, the maximum constraint violation of the current iteration is recorded to be the threshold of the next iteration. On the other hand, if the maximum constraint violation of the current iteration has not reduced enough, we update the penalty parameter that is 10 time more than the previous one $\sigma = 10 \times \sigma$ to enforce the constraints

Step 4: Stop criteria checking: In this step, we will utilize two thresholds to decide whether the iteration stops or not. The first criterion is the maximum constraint violation $V_{threshold}$. The second one is the maximum loop number. If the current maximum constraint violation less than the pre-defined threshold $V \leq V_{threshold}$, then, we terminate the iteration. Otherwise, if the loop index l is more than the pre-defined maximum loop limit $l_{\max\_iter}$, we also end the iteration. The final optimization results will be derived after the end of the iterative calculation.

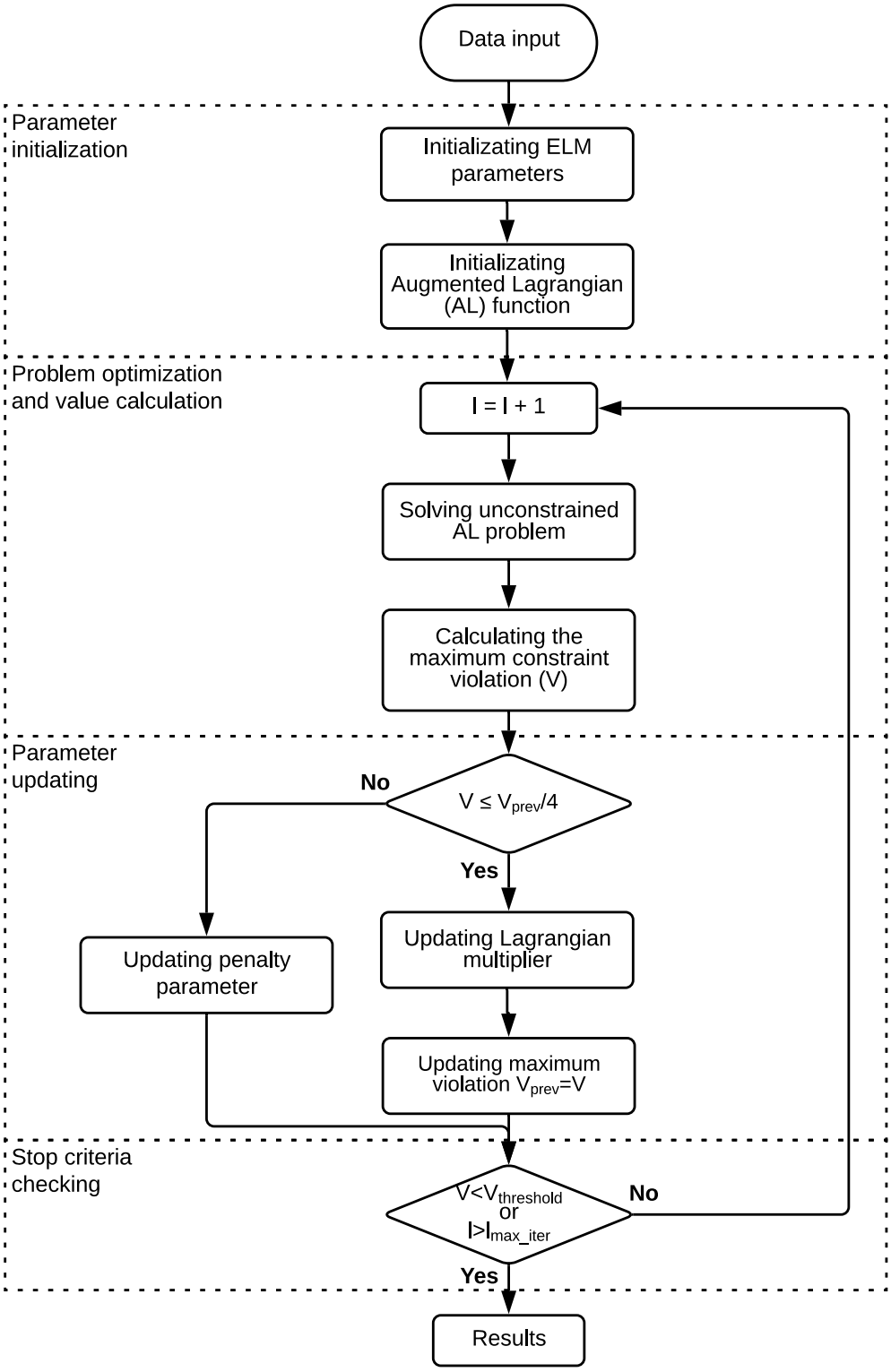


Figure 4.2 Flowchart to solve the proposed optimal prediction interval problem.

In order to reduce computation time and derive stable prediction interval forecasts, we also propose a parallel computation structure that is shown in Figure 4.3. To avoid wasting time to share information between different modules, each computation module includes the whole optimization procedures that are shown in Figure 4.2. The final prediction intervals are the ensemble of the upper and lower bounds of the prediction intervals that are derived by each computation module.

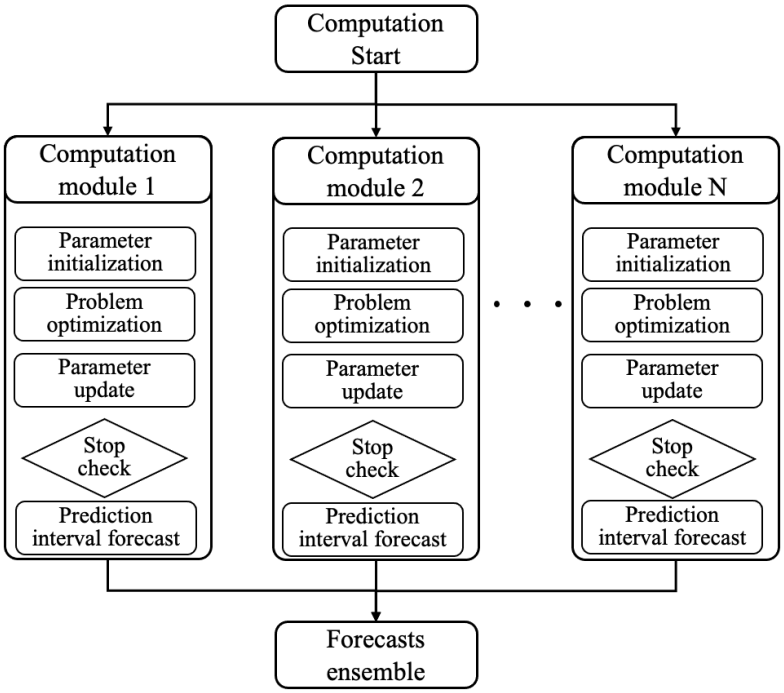


Figure 4.3 Parallel optimal prediction interval computation structure.

4.3 Case studies

In this subsection, we use numeric results of case studies to show the effectiveness of our proposed optimal prediction interval method.

1. As electricity price data are heteroscedastic so that the distribution is not pre-fixed, to show our proposed non-parametric method is able to provide accurate predictions, we

compare our forecast results with the prediction intervals derived from the method that assumes price distribution in advanced.

2. To demonstrate that the prediction intervals derived from the proposed method are sharper while meet reliability requirement, we compare our forecast results with the prediction intervals forecasted by using the neural network with the bootstrap, which both types of methods are non-parametric and can learn non-linear relationships.
3. To show method versatility, instead of just using data from the Pennsylvania-New Jersey-Maryland Interconnection (PJM), we also adopt our proposed method on different electricity markets, such as the Australian Energy Market (AEM) or the Ontario electricity market, and compare forecast results with other references.

4.3.1 Proposed and Parametric Methods Forecasts Comparison

In this section, we adopt our proposed method to forecast the prediction interval of the PJM day-ahead locational marginal price (LMP), where the data are available in [30].

Figure 4.4 shows the LMP probability density function (PDF) of each season where the covered areas are filled with different colors. In this figure, we can tell that the day-ahead electricity prices of the PJM are not symmetrically distributed. Majority of electricity prices are located around 10$/MWh to 40$/MWh. And only a few electricity prices are higher than 50$/MWh. Therefore, the normal distribution is an error-prone assumption to describe the electricity price distribution. This phenomenon also justifies the benefits of choosing a non-parametric method to forecast the prediction interval without defining the distribution of price data in advance. Besides, from the Figure 4.4, we can observe that the distributions of the electricity price within each season have different shapes. Therefore, we adopt different models to forecast the price prediction intervals of each season.

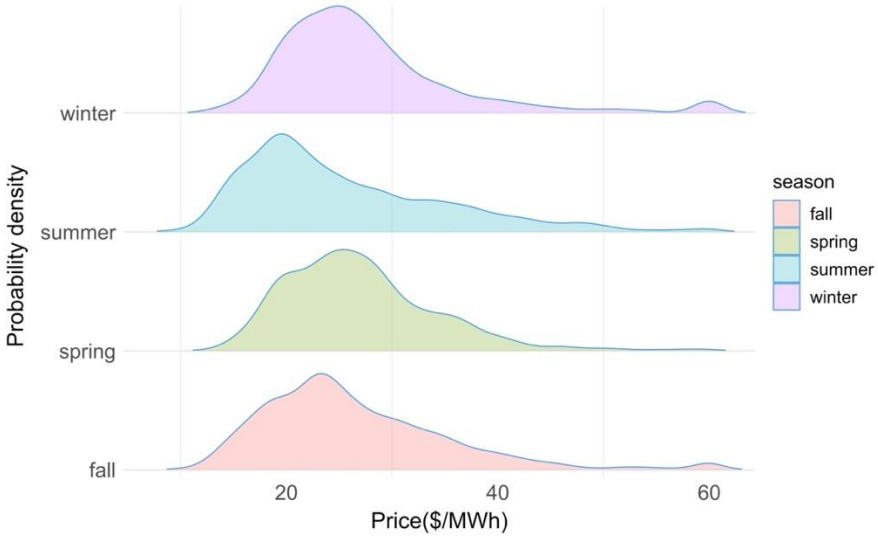


Figure 4.4 PJM LMP probability density function of each season in 2019.

To compare the forecast performance of non-parametric and parametric methods, we demonstrate the prediction intervals derived from our proposed method and the ARIMA method that assumes the residuals follow normal distribution. To visualize our forecast results, in Figure 4.5 – 4.8, we plot the prediction intervals of the PJM market in the spring (March to May), summer (June to August), fall (September to November), and winter (December to February) in 2019, respectively. As our focus is on the day-ahead electricity price prediction, the forecast horizon is the next 24 hours. Without loss of generality, the nominal coverage probability of the prediction interval is set to be 90%. From plots, we can observe that the day-ahead electricity prices of the PJM are heteroscedastic. Although the price distribution in each season has its own characteristic, the record day-ahead prices of different seasons are bounded by the prediction intervals derived from our proposed method that demonstrate our predictions are reliable. Besides, the width between the upper and lower bound of prediction interval are narrow that shows the sharpness of our forecasts.

Table 4.1 Prediction intervals comparison between proposed method and ARIMA method

Season	PICP (%)		ACE (%)		PIAW ($/MWh)	
	Proposed	ARIMA	Proposed	ARIMA	Proposed	ARIMA
Winter	90.42	97.71	0.42	7.71	11.75	32.27
Spring	90.53	88.25	0.53	-1.75	9.29	11.15
Summer	93.23	86.58	3.23	-3.41	9.09	13.23
fall	92.71	86.04	2.71	-3.96	15.62	16.97

The detailed statistics related to the forecast results are listed in Table 4.1. To compare prediction interval reliability, from Table 4.1, we can tell that the prediction interval coverage probability derived from our proposed method is closer to the pre-defined nominal confidence than the ARIMA method. This situation is also reflected in the ACE metric that the deviations of the PICP from the PINC of the proposed method are between 0.42% to 3.23% where the ACE values of the ARIMA method range from -3.96% to 7.71%. Besides, to compare forecast results sharpness, the PIAW value of our proposed method are lower than the PIAW value of the ARIMA method. As narrow prediction intervals could give market participants more confidence in decision making, our proposed method is able to not only provide reliable forecast results but also prediction intervals with a narrow width.

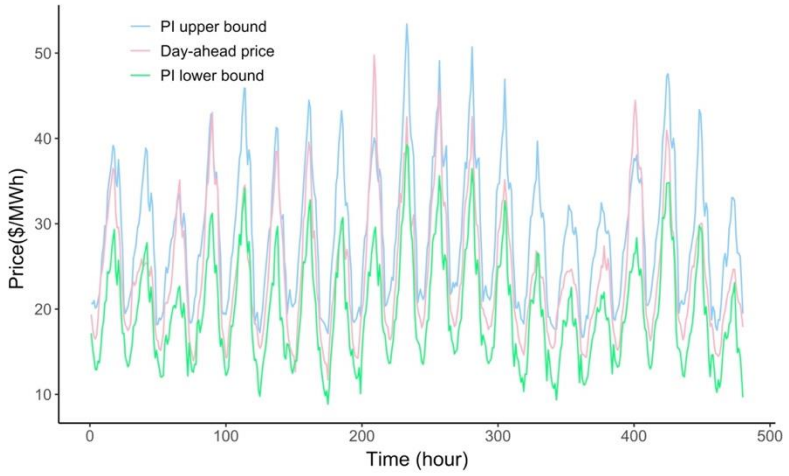


Figure 4.5 Prediction interval of the day-ahead electricity price with 90% nominal coverage probability in spring 2019.

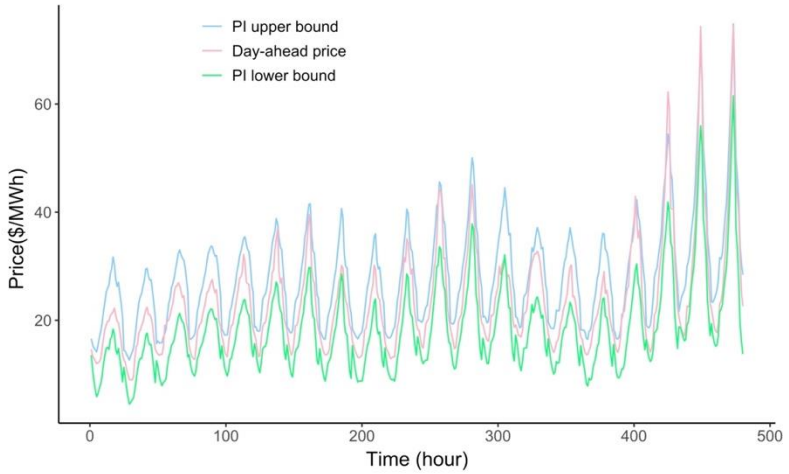


Figure 4.6 Prediction interval of the day-ahead electricity price with 90% nominal coverage probability in summer 2019.

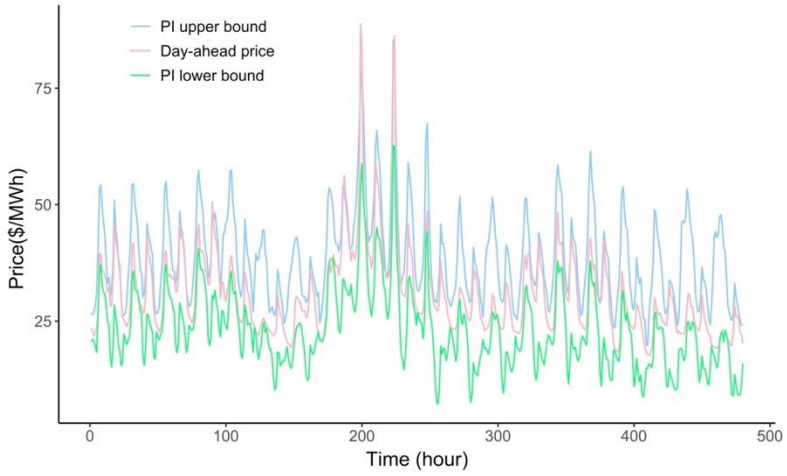


Figure 4.7 Prediction interval of the day-ahead electricity price with 90% nominal coverage probability in fall 2019.

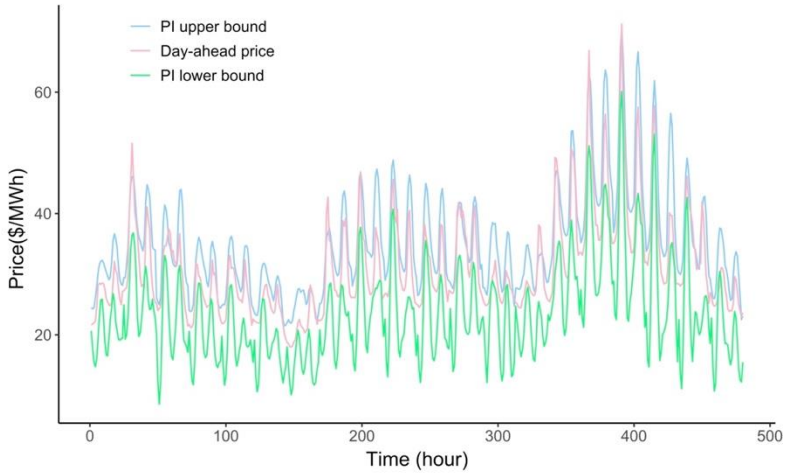


Figure 4.8 Prediction interval of the day-ahead electricity price with 90% nominal coverage probability in winter 2019.

4.3.2 Proposed and Neural Network Forecasts Comparison

To further evaluate our method, we compare the prediction intervals derived from our method with the forecast results predicted by using the neural network (NN) approach with bootstrap strategy. The reasons to choose the neural network method as another benchmark are two-fold. The first reason is that both our proposed method and the neural network method are

able to learn non-linear relationship within price data. Besides, by adopting the bootstrap strategy, the neural network method is also a type of non-parametric method to generate prediction intervals without assuming price data distribution in advance. However, the difference between our proposed method and the neural network method is that we take prediction interval sharpness requirement into consideration. The detailed statistics related to the prediction intervals are provided in Table 4.2.

To give a comprehensive comparison, we evaluate the forecast results from the reliability and sharpness perspectives. From Table 4.2, as the ACE value of the proposed method is less than 5% whereas the ACE value of the neural network are around 10%, we can tell that the coverage of prediction interval derived from our proposed method are closer to the pre-defined requirement than results predicted by using the neural network method. From sharpness perspective, from Table 4.2, we can tell that the prediction intervals of our proposed method are tighter the neural network in each season. Another interesting observation is that, by comparing the PIAW values from Table 4.1 and 4.2, we can tell that although the prediction intervals of neural network are wider than our proposed method, their width is still narrower than forecast results of the ARIMA method. This results further demonstrate the benefits of non-parametric method.

Table 4.2 Prediction intervals comparison between proposed method and neural network method

Season	PICP (%)		ACE (%)		PIAW ($/MWh)	
	Proposed	ARIMA	Proposed	ARIMA	Proposed	ARIMA
Winter	90.42	85	0.42	5.00	11.75	18.91
Spring	90.53	77.46	0.53	-12.54	9.29	10.01
Summer	93.23	75.91	3.23	-14.09	9.09	10.28
fall	92.71	79.38	2.71	-10.62	15.62	15.72

4.3.3 Overall Comparison

To demonstrate the versatility of the proposed method, we provide more detailed forecast results related to the PJM market including the prediction intervals with 95% and 99% nominal coverage probability. Besides, we also test our method using data from electricity markets, such as the Australian Energy Market (AEM) and Independent Electricity System Operator (IESO) in Canada, and compare our prediction results with references [5] and [38].

For the PJM electricity market, from Tables 4.3, the proposed forecast method provides reliable results as the prediction interval coverages measured by PICP of the proposed method are close to the nominal coverage probability. And all of the prediction interval forecast deviations measure by the absolute ACE are smaller than 3% where the largest ACE values of the 95% and 99% nominal coverage are 2.58% and 1.21%, respectively. Besides, by comparing the 95% and 99% PIAW values, it shows that although prediction interval coverage increases to accommodate more extreme prices, the width of prediction interval does not increase dramatically. For instance, the PIAW of confidence level 95% and 99% are 10.86$/MWh and 15.11$/MWh in winter.

Table 4.3 PJM prediction intervals with 95% and 99% nominal coverage probability

Season	PICP (%)		ACE (%)		PIAW ($/MWh)	
	PI 95	PI 99	PI 95	PI 99	PI 95	PI 99
Winter	92.92	98.13	-2.08	-0.87	10.86	15.11
Spring	92.99	98.86	-2.01	-0.13	10.20	15.65
Summer	96.48	97.79	1.48	-1.21	11.41	15.40
fall	97.58	99.46	2.58	0.46	20.59	32.35

The prediction results related to the Australia electricity market AEMO are shown in Table 4.4 and 4.5 that related to 90% and 99% prediction interval forecasts, respectively. Note

that four seasons in Australia are winter (June–August), spring (September–November), summer (December–February), and autumn (March–May). From Table 4.4 and 4.5, we can tell that prediction interval proposed by the proposed are also close to the nominal coverage probability. And compared with PICP values from the reference, the proposed method normally will provide reliable results. Besides, by comparing the PIAW values, we can tell that our forecast results avoid quantile crossing situation as the width of prediction interval is increasing with respect to the probability coverage increase. For example, the PIAW of confidence level 90% and 99% are 34.53$/MWh and 62.59$/MWh in spring.

Table 4.4 AEMO 90% prediction intervals comparison

Season	PICP (%)		ACE (%)		PIAW ($/MWh)
	Proposed	Ref [5]	Proposed	Ref [5]	Proposed
Winter	89.97	89.88	-0.03	-0.12	19.55
Spring	90.14	92.36	0.14	2.26	34.53
Summer	94.46	89.88	4.64	-0.12	39.81
fall	90.47	89.29	0.47	-0.71	46.78

Table 4.5 AEMO 99% prediction intervals comparison

Season	PICP (%)		ACE (%)		PIAW ($/MWh)
	Proposed	Ref [5]	Proposed	Ref [5]	Proposed
Winter	99.24	97.92	0.24	-1.08	49.03
Spring	98.95	97.32	-0.05	-1.68	62.59
Summer	98.86	97.62	-0.14	-1.38	218.89
fall	98.81	98.21	-0.19	-0.79	115.2

Table 4.6 IESO 90% prediction intervals comparison

Season	PICP (%)		ACE (%)		PIAW ($/MWh)	
	Proposed	Ref [38]	Proposed	Ref [38]	Proposed	Ref [38]
Winter	92.95	90.48	2.95	0.48	16.47	37.92
Spring	91.11	92.26	1.11	2.26	26.45	54.85
Summer	87.77	88.10	-2.23	-1.90	27.48	8.45
fall	90.10	93.45	0.10	3.45	20.80	42.70

Table 4.6 shows 90% prediction interval forecasts of the Ontario electricity markets IESO where the Ontario electricity market has some of most volatile prices in North America [59]. From Table 4.6, we can tell that the volatile of electricity price does have impact on prediction interval forecast. Some of the ACE values of 90% nominal coverage forecasts are more than 1% in the PJM or AEMO markets while most of ACE are more than 1% in the IESO market. Compared with the reference, although the electricity prices are very volatile, the proposed method could provide reliability results while ensure sharpness requirements that most of the interval widths of the proposed method are narrow than the reference results.

4.4 Conclusions

As uncertainty is inevitable in electricity price forecasts, power market participants often use probabilistic forecasts to quantify the uncertainty. Although wide prediction interval could reach a high reliability level, due to overlook sharpness, market participants are prone to make costly and conservative decisions.

In this section, we propose an optimal electricity price prediction interval method that considers both reliability and sharpness. In order to ensure that nonlinear relationships between the predictor and the target variable are identified, the extreme learning machine method is adopted. Besides, the upper and lower bound of the prediction interval is derived from quantile regressions without assuming electricity price distribution.

CHAPTER V

PRELIMINARY CONCLUSION AND FUTURE WORK

As all market participants require electricity price forecasts to hedge risks, reduce cost, and make profits, therefore, the more accurate predictions they can get, the better decisions they will derive. According to literature reviews and numerical results, for point forecast, by analyzing price data within frequency domain and utilizing the frequencies information, we could get accurate price forecast results under different circumstances. Besides, the price spike oversampling method can resolve the spike forecast difficulties caused by real data shortage. And for probabilistic forecast, to improve prediction results quality, the optimal prediction interval method not only considers the reliability requirement but also considers the sharpness criterion. To avoid the influence of error-prone price distribution assumption, we adopt the quantile regression in our study. Besides, the ELM method is adopted in our method to capture the nonlinear relationships within price data and ensure our method versatility.

Based on the study and results shown in this proposal, we are going to conduct the following work in the near future. Forecast combination is a widely used as method to improve point forecast accuracy. However, this concept has not been formally exploited in the probabilistic forecasts. To further improve forecast quality, we are planning to develop an optimal combination of electricity price probabilistic forecasts technique that further improve the quality of prediction interval forecasts. And to ensure the ability of the proposed method to work under different circumstances, we are going to use the proposed method to forecast electricity

prices of different seasons and from different markets. Besides, to show the benefits of our method, we also decide to compare our forecast results with the prediction results derived from some benchmark methods.

REFERENCES

1. W. Tang, R. Rajagopal, K. Poolla and P. Varaiya, "Model and data analysis of two-settlement electricity market with virtual bidding," in *2016 IEEE 55th Conference on Decision and Control* (CDC), 2016, pp. 6645-6650, doi: 10.1109/CDC.2016.7799292.

2. A. Bello, D. W. Bunn, J. Reneses and A. Muñoz, "Medium-Term Probabilistic Forecasting of Electricity Prices: A Hybrid Approach," in *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 334-343, Jan. 2017, doi: 10.1109/TPWRS.2016.2552983.

3. F. Ziel and R. Steinert, “Probabilistic mid- and long-term electricity price forecasting,”

   Renewable and Sustainable Energy Reviews, vol. 94, pp. 251–266, 2018.

4. R. C. Garcia, J. Contreras, M. van Akkeren and J. B. C. Garcia, "A GARCH forecasting model to predict day-ahead electricity prices," in *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 867-874, May 2005.

5. X. Chen, Z. Y. Dong, K. Meng, Y. Xu, K. P. Wong and H. W. Ngan, "Electricity Price Forecasting With Extreme Learning Machine and Bootstrapping," in *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 2055-2062, Nov. 2012.

6. F. J. Nogales, J. Contreras, A. J. Conejo and R. Espinola, "Forecasting next-day electricity prices by time series models," in *IEEE Transactions on Power Systems*, vol. 17, no. 2, pp. 342-348, May 2002.

7. J. H. Zhao, Z. Y. Dong, Z. Xu and K. P. Wong, "A Statistical Approach for Interval Forecasting of the Electricity Price," in *IEEE Transactions on Power Systems*, vol. 23, no. 2, pp. 267-276, May 2008.

8. S. Chai, Z. Xu and Y. Jia, "Conditional Density Forecast of Electricity Price Based on Ensemble ELM and Logistic EMOS," in *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3031-3043, May 2019.

9. A. T. Lora, J. M. R. Santos, A. G. Exposito, J. L. M. Ramos and J. C. R. Santos, "Electricity Market Price Forecasting Based on Weighted Nearest Neighbors Techniques," in *IEEE Transactions on Power Systems*, vol. 22, no. 3, pp. 1294-1301, Aug. 2007.

10. P. Mandal, T. Senjyu, N. Urasaki, T. Funabashi and A. K. Srivastava, "A Novel Approach to Forecast Electricity Price for PJM Using Neural Network and Similar Days Method," in *IEEE Transactions on Power Systems*, vol. 22, no. 4, pp. 2058-2065, Nov. 2007.

11. G. P. Zhang, “Time series forecasting using a hybrid ARIMA and neural network model,” in *Neurocomputing*, vol. 50, pp. 159–175, 2003.

12. L. Wu and M. Shahidehpour, "A Hybrid Model for Day-Ahead Price Forecasting," in *IEEE Transactions on Power Systems*, vol. 25, no. 3, pp. 1519-1530, Aug. 2010.

13. N. Amjady and F. Keynia, “Electricity market price spike analysis by a hybrid data model and feature selection technique,” in *Electric Power Systems Research*, vol. 80, no. 3, pp. 318–327, 2010.

14. H. Chitsaz, P. Zamani-Dehkordi, H. Zareipour and P. P. Parikh, "Electricity Price Forecasting for Operational Scheduling of Behind-the-Meter Storage Systems," in *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6612-6622, Nov. 2018.

15. J. H. Zhao, Z. Y. Dong, X. Li and K. P. Wong, "A Framework for Electricity Price Spike Analysis With Advanced Data Mining Methods," in *IEEE Transactions on Power Systems*, vol. 22, no. 1, pp. 376-385, Feb. 2007.

16. J. H. Zhao, Z. Y. Dong and X. Li, "Electricity market price spike forecasting and decision making," in *IET Generation, Transmission & Distribution*, vol. 1, no. 4, pp. 647-654, July 2007.

17. Lu, X., Dong, Z.Y., and Li, X., “Electricity market price spike forecast with data mining techniques”, in *Electr. Power Syst. Res.*, 2005, 73, (1), pp. 19 – 29.

18. Zhao, J.H., Dong, Z.Y., Li, X., and Wong, K.P., “A general method for electricity market price spike analysis”, in *IEEE Power Engineering Society General Meeting*, June 2005, pp. 563–570.

19. N. Amjady and F. Keynia, “A new prediction strategy for price spike forecasting of day-ahead electricity markets,” in *Applied Soft Computing*, vol. 11, no. 6, pp. 4246–4256, 2011.

20. H. S. Sandhu, L. Fang, and L. Guan, “Forecasting day-ahead price spikes for the Ontario electricity market”, in *Electric Power Systems Research*, vol. 141, pp. 450–459, 2016.

21. T. Christensen, A. Hurn, and K. Lindsay, “Forecasting spikes in electricity prices,” in *International Journal of Forecasting*, vol. 28, no. 2, pp. 400–411, 2012.

22. V. Liepins, “Extended Fourier analysis of signals,” arXiv:1303.2033v9 [cs.DS], Mar. 2019.

23. R. J. Hyndman, and G. Athanasopoulos, *Forecasting: principles and practice*, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2.

24. K. Dragomiretskiy and D. Zosso, "Variational Mode Decomposition," in *IEEE Transactions on Signal Processing*, vol. 62, no. 3, pp. 531-544, Feb.1, 2014.

25. H. Cao, X. Li, D. Y. Woon and S. Ng, "Integrated Oversampling for Imbalanced Time Series Classification," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 12, pp. 2809-2822, Dec. 2013.

26. R. Ribeiro, "Utility-based Regression". Ph.D. dissertation, Dep. C.S., Univ. of Porto, Porto, Portugal, 2011.

27. L. Torgo, R. P. Ribeiro, B. Pfahringer, and P. Branco, "SMOTE for Regression," *Progress in Artificial Intelligence*, pp. 378–389, 2013.

28. *UCI Machine Learning Repository*: Synthetic Control Chart Time Series Data Set. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Synthetic Control Chart Time Series. [Accessed: 02-Dec-2019].

29. X. Jiang, "Linear subspace learning-based dimensionality reduction," *IEEE Signal Processing Magazine*, vol. 28(2), Mar. 2011, pp. 16-26.

30. PJM Web Site. [Online]. Available: http://www.pjm.com.

31. MISO Web Site. [Online]. Available: https://www.misoenergy.org/.

32. ISO-NE Web Site. [Online]. Available: https://www.iso-ne.com/.

33. NYISO Web Site. [Online]. Available: https://www.nyiso.com/.

34. SPP Web Site. [Online]. Available: https://marketplace.spp.org.

35. Market Operator of the Electricity Market of Mainland Spain, OMEL. [Online]. Available: https://www.omie.es/en/sobre-nosotros.

36. P. Pinson and G. Kariniotakis, "Conditional Prediction Intervals of Wind Power Generation," in *IEEE Transactions on Power Systems*, vol. 25, no. 4, pp. 1845-1856, Nov. 2010, doi: 10.1109/TPWRS.2010.2045774.

37. G. Sideratos and N. D. Hatziargyriou, "Probabilistic Wind Power Forecasting Using Radial Basis Function Neural Networks," in *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 1788-1796, Nov. 2012, doi: 10.1109/TPWRS.2012.2187803.

38. N. A. Shrivastava, A. Khosravi and B. K. Panigrahi, "Prediction Interval Estimation of Electricity Prices Using PSO-Tuned Support Vector Machines," in *IEEE Transactions on Industrial Informatics*, vol. 11, no. 2, pp. 322-331, April 2015, doi: 10.1109/TII.2015.2389625.

39. C. Zhao, C. Wan and Y. Song, "An Adaptive Bilevel Programming Model for Nonparametric Prediction Intervals of Wind Power Generation," in *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 424-439, Jan. 2020, doi: 10.1109/TPWRS.2019.2924355.

40. C. Wan, Z. Xu, P. Pinson, Z. Y. Dong and K. P. Wong, "Optimal Prediction Intervals of Wind Power Generation," in *IEEE Transactions on Power Systems*, vol. 29, no. 3, pp. 1166-1174, May 2014, doi: 10.1109/TPWRS.2013.2288100.

41. A. J. Cannon, “Quantile regression neural networks: Implementation in R and application to precipitation downscaling,” *Computers & Geosciences*, vol. 37, no. 9, pp. 1277–1284, 2011.

42. A. J. Cannon, “Non-crossing nonlinear regression quantiles by monotone composite quantile regression neural network, with application to rainfall extremes,” *Stochastic Environmental Research and Risk Assessment*, vol. 32, no. 11, pp. 3207–3225, 2018.

43. A. Khosravi, S. Nahavandi, and D. Creighton, “Quantifying uncertainties of neural network-based electricity price forecasts,” *Applied Energy*, vol. 112, pp. 120–129, 2013.

44. P. Mandal, A. K. Srivastava, and J.-W. Park, “An effort to optimize similar days parameters for ann-based electricity price forecasting,” *IEEE Transactions on Industry Applications*, vol. 45, no. 5, pp. 1888–1896, 2009.

45. C. Wan, J. Wang, J. Lin, Y. Song and Z. Y. Dong, "Nonparametric Prediction Intervals of Wind Power via Linear Programming," in *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 1074-1076, Jan. 2018, doi: 10.1109/TPWRS.2017.2716658.

46. G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, “Extreme learning machine: Theory and applications,” *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, Dec. 2006.

47. G. Huang, H. Zhou, X. Ding and R. Zhang, "Extreme Learning Machine for Regression and Multiclass Classification," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513-529, April 2012, doi: 10.1109/TSMCB.2011.2168604.

48. M. Kaj, T. Ole, and N. H. Bruun, *Optimization with Constraints, 2nd ed.*, 2004. [Online]. Available: http://www2.imm.dtu.dk/pubdb/p.php?4213. [Accessed: 15-Jul-2021].

49. R. Fletcher, *Practical methods of optimization*. Chichester: Wiley, 2000.

50. Y. Ji, R. J. Thomas and L. Tong, "Probabilistic Forecasting of Real-Time LMP and Network Congestion," in *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 831-841, March 2017, doi: 10.1109/TPWRS.2016.2592380.

51. L. Hao and D. Q. Naiman, *Quantile regression*. Thousand Oaks, CA: Sage Publications, 2007.

52. W. Wang and N. Yu, "A Machine Learning Framework for Algorithmic Trading with Virtual Bids in Electricity Markets," *2019 IEEE Power & Energy Society General Meeting (PESGM)*, 2019, pp. 1-5, doi: 10.1109/PESGM40551.2019.8973750.

53. C. Wan, J. Lin, J. Wang, Y. Song and Z. Y. Dong, "Direct Quantile Regression for Nonparametric Probabilistic Forecasting of Wind Power Generation," in *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2767-2778, July 2017, doi: 10.1109/TPWRS.2016.2625101.

54. M. Zhou, Z. Yan, Y. X. Ni, G. Li, and Y. Nie, "Electricity price forecasting with confidence-interval estimation through an extended ARIMA approach," *IEE Proceedings - Generation, Transmission and Distribution*, vol. 153, no. 2, p. 187, 2006.

55. R. Weron and A. Misiorek, "Forecasting spot electricity prices: A comparison of parametric and semiparametric time series models," *Int. J. Forecast.*, vol. 24, no. 4, pp. 744–763, 2008.

56. Nix, D. and A. Weigend. "Learning Local Error Bars for Nonlinear Regression." *NIPS*, 1994.

57. C. Wan, Z. Xu, Y. Wang, Z. Y. Dong and K. P. Wong, "A Hybrid Approach for Probabilistic Forecasting of Electricity Price," in *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 463-470, Jan. 2014.

58. C. Wan, M. Niu, Y. Song and Z. Xu, "Pareto Optimal Prediction Intervals of Electricity Price," in *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 817-819, Jan. 2017, doi: 10.1109/TPWRS.2016.2550867

59. H. Zareipour, A. Janjani, H. Leung, A. Motamedi and A. Schellenberg, "Classification of Future Electricity Market Prices," in *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 165-173, Feb. 2011, doi: 10.1109/TPWRS.2010.2052116.

60. J. Lago, G. Marcjasz, B. De Schutter, and R. Weron, "Forecasting Day-Ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark," *Applied Energy*, vol. 293, p. 116983, 2021.

61. J. Xie, T. Hong, T. Laing, and C. Kang, "On normality assumption in residual simulation for probabilistic load forecasting," *IEEE Transactions on Smart Grid*, vol. 8, no. 3, pp. 1046–1053, 2017.

62. T. Hong and S. Fan, "Probabilistic Electric Load Forecasting: A tutorial review," *International Journal of Forecasting*, vol. 32, no. 3, pp. 914–938, 2016.

63. Y. Cao and Z.-J. M. Shen, "Quantile forecasting and data-driven inventory management under nonstationary demand," *Operations Research Letters*, vol. 47, no. 6, pp. 465–472, 2019.

64. H. C. Wu, S. C. Chan, K. M. Tsui, and Y. Hou, “A new recursive dynamic factor analysis for point and interval forecast of electricity price,” *IEEE Transactions on Power Systems*, vol. 28, no. 3, pp. 2352–2365, 2013.

65. M. Rafiei, T. Niknam, and M.-H. Khooban, “Probabilistic forecasting of hourly electricity price by generalization of elm for usage in improved wavelet neural network,” *IEEE Transactions on Industrial Informatics*, vol. 13, no. 1, pp. 71–79, 2017.

66. R. Tahmasebifar, M. K. Sheikh-El-Eslami, and R. Kheirollahi, “Point and interval forecasting of real-time and Day-ahead electricity prices by a novel hybrid approach,” *IET Generation, Transmission & Distribution*, vol. 11, no. 9, pp. 2173–2183, 2017.

67. L. H. Kaack, J. Apt, M. G. Morgan, and P. McSharry, “Empirical prediction intervals improve energy forecasting,” *Proceedings of the National Academy of Sciences*, vol. 114, no. 33, pp. 8752–8757, 2017.

68. P. Mandal, A. U. Haque, J. Meng, A. K. Srivastava, and R. Martinez, “A novel hybrid approach using wavelet, Firefly algorithm, and Fuzzy Artmap for day-Ahead Electricity Price forecasting,” *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1041–1051, 2013.

69. F. Serinaldi, “Distributional modeling and short-term forecasting of electricity prices by generalized additive models for location, scale and shape,” *Energy Economics*, vol. 33, no. 6, pp. 1216–1226, 2011.

70. A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, “Comprehensive review of neural network-based prediction intervals and new advances,” *IEEE Transactions on Neural Networks*, vol. 22, no. 9, pp. 1341–1356, 2011.

71. Z. Xiao, “Time series quantile regressions,” *Time Series Analysis: Methods and Applications*, pp. 213–257, 2012.

72. R. Cao, J. Hart, and ángeles Saavedra, “Nonparametric maximum likelihood estimators for AR and Ma Time series,” *Journal of Statistical Computation and Simulation*, vol. 73, no. 5, pp. 347–360, 2003.

73. Y. Wang, Q. Chen, N. Zhang, and Y. Wang, “Conditional residual modeling for Probabilistic Load forecasting,” *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 7327–7330, 2018.

74. J.-F. Toubeau, J. Bottieau, F. Vallee, and Z. De Greve, “Deep learning-based multivariate probabilistic forecasting for short-term scheduling in power markets,” *IEEE Transactions on Power Systems*, vol. 34, no. 2, pp. 1203–1215, 2019.

75. C. Zhao, C. Wan, Y. Song, and Z. Cao, “Optimal nonparametric prediction intervals of Electricity Load,” *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 2467–2470, 2020.

76. M. Rafiei, T. Niknam, J. Aghaei, M. Shafie-Khah, and J. P. Catalao, “Probabilistic load forecasting using an improved wavelet neural network trained by Generalized Extreme Learning Machine,” *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6961–6971, 2018.

77. B. Liu, J. Nowotarski, T. Hong, and R. Weron, “Probabilistic load forecasting via quantile regression averaging on sister forecasts,” *IEEE Transactions on Smart Grid*, pp. 1–1, 2015.

78. K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, “Short-term load forecasting with deep residual networks,” *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 3943–3952, 2019.

79. J. Xie and T. Hong, “Temperature scenario generation for probabilistic load forecasting,” *IEEE Transactions on Smart Grid*, pp. 1–1, 2016.

80. M. Sun, T. Zhang, Y. Wang, G. Strbac, and C. Kang, “Using bayesian deep learning to capture uncertainty for residential net load forecasting,” *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 188–201, 2020.

81. J. Xie and T. Hong, “Variable selection methods for probabilistic load forecasting: Empirical evidence from seven states of the United States,” *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6039–6046, 2018.

82. W. Zhang, H. Quan, and D. Srinivasan, “An improved quantile regression neural network for probabilistic load forecasting,” *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4425–4434, 2019.

83. J. Contreras, R. Espinola, F. J. Nogales, and A. J. Conejo, “Arima models to predict next-day electricity prices,” IEEE Transactions on Power Systems, vol. 18, no. 3, pp. 1014–1020, 2003.

84. W. Zhang, H. Quan, O. Gandhi, R. Rajagopal, C.-W. Tan, and D. Srinivasan, “Improving probabilistic load forecasting using quantile regression NN with skip connections,” *IEEE Transactions on Smart Grid*, vol. 11, no. 6, pp. 5442–5450, 2020.

85. N. Amjady and F. Keynia, “Day-ahead price forecasting of electricity markets by mutual information technique and cascaded neuro-evolutionary algorithm,” *IEEE Transactions on Power Systems*, vol. 24, no. 1, pp. 306–318, 2009.