A random forest method for real-time price forecasting in New York electricity market

Conference Paper · July 2014

DOI: 10.1109/PESGM.2014.6939932

CITATIONS
21

READS
2,259

5 authors, including:

Jie Mei
Massachusetts Institute of Technology
16 PUBLICATIONS **69** CITATIONS
SEE PROFILE

Dawei He
Georgia Institute of Technology
25 PUBLICATIONS **273** CITATIONS
SEE PROFILE

R.G. Harley
Georgia Institute of Technology
596 PUBLICATIONS **15,902** CITATIONS
SEE PROFILE

A Random Forest Method for Real-Time Price Forecasting in New York Electricity Market

Jie Mei, Dawei He, Ronald Harley and Thomas Habetler
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, USA

Guannan Qu
Department of Electrical Engineering
Tsinghua University
Beijing, China

Abstract— **This paper mainly focuses on the real-time price forecasting in New York electricity market through random forest. Accurate forecasting is regarded as the most practical way to win power bid in today's highly competitive electricity market. Comparing with existing price forecasting methods, random forest, as a newly introduced method, will provide a price probability distribution, which will allow the users to estimate the risks of their bidding strategy and also making the results helpful for later industrial using. Furthermore, the model can adjust to the latest forecasting condition, i.e. the latest climatic, seasonal and market condition, by updating the random forest parameters with new observations. This adaptability avoids the model failure in a climatic or economic condition different from the training set. A case study in New York HUD VL area is presented to evaluate the proposed model.**

Index Terms—**random forest, electricity price, NYISO, electricity market.**

I. INTRODUCTION

IN New York Independent System Operator (NYISO), a hybrid PoolCo/Bilateral Contracts model is applied. In the Bilateral part, customer can choose to sign the bilateral contracts with suppliers, in which case, they choose to accept power at spot market price. The day-ahead electricity price is determined by bilateral contract. While for any customer who do not choose to accept power at the spot market price, or in another word, do not choose to sign the day-ahead contract, they would be allowed to negotiate a power supply agreement directly with suppliers. The real-time electricity price is determined in this case. The PoolCo model would serve all participants (buyers and sellers) who choose not to sign bilateral contracts. In this model, electric power sellers/buyers submit bids to the pool for the amounts of power that they are willing to trade in the market. [1] In nowadays' electricity market, this biding and clearing process repeats every five minutes. Thus short-term price forecasting in competitive electricity markets is critical for consumers and producers in planning their operations and managing their price risk. [2] We propose a new random forest forecasting method along with ARMA and ANN for benchmark here.

The current methods can be divided into the following aspects. 1) Time Series: ARIMA and GARH. 2) Machine Learning: ANN, SVM

As early as 2003, Contreras [3] published an ARIMA next-day electricity price forecasting method. It simply used the historical price data to train the ARIMA model and found the general tendency of next-day price. Contreras tested his model in California electricity market, the MAPE is around 15%. And the model forecasted the price every hour, which is not suitable for a competitive electricity market as todays' NYISO.

Another time series approach, which is very similar to ARMA, based on the Generalized Autoregressive Conditional Heteroskedastic (GARCH) was proposed by R. Garcia [4] in 2005. The model was tested in the Spain and California deregulated electricity market and the MAPE is about 12.5%.

One year later, in 2006, Amjady [5] proposed a new fuzzy neural network for short-term price forecasting of electricity markets. This fuzzy neural network has inter-layer and feed-forward architecture with a new hyper-cubic training mechanism. The proposed method predicts hourly market-clearing prices for the day-ahead electricity markets and it was tested in Spanish electricity market. The MAPE is about 11.4% on average.

Fan and Mao [6] proposed a novel method of forecasting short-term electricity price based on a two-stage hybrid network of self-organized map (SOM) and support-vector machine (SVM). In the first stage, a SOM network is applied to cluster the input-data set into several subsets in an unsupervised manner. Then, a group of SVMs is used to fit the training data of each subset in the second stage in a supervised way. To confirm its effectiveness, the proposed model has been trained and tested on the data of historical energy prices from the New England electricity market. The MAPE is 10.24%.

In 2009, Mandal [7] proposed an improved ANN electricity price forecasting method in which he added a sensitive analysis of similar day parameters to increase the model's accuracy. The improved model was tested in Pennsylvania–New Jersey–Maryland (PJM) electricity market and the MAPE is around 11%. Also, this model conducted the forecasting in hourly base.

The topic of real-time electricity price forecasting has been discussed extensively in recent years. Much effort has been put into this area from electricity sellers and buyers for the purpose of getting the best bidding strategy. However, the previous proposed methods have several bottlenecks. First, almost all of them were trying to predict the electricity price every thirty minutes or every hour rather than every five minutes we want. Second, simple price prediction is not so

978-1-4799-6415-4/14/$31.00 ©2014 IEEE

helpful compared with price probability distribution, which can help the sellers/buyers estimate the risk of their bidding decisions. In the price probability distribution, they can know the probability for a specific electricity price. Third, the previous forecasting models are not updatable. The market and climate are changing, which means we need a model that can automatically adjust to the latest market and climatic condition.

A Random Forest based adaptive forecasting framework is proposed in this paper. By utilizing its bootstrap distribution, it can provide a confidence interval attached to the prediction. Furthermore, the random forest adjust to the latest forecasting condition, i.e. the latest climatic, seasonal and market condition, by updating the random forest parameters with new observations. This adaptability avoids the model failure in a climatic or economic condition different form the training set. A case study is conducted to prove the validity of the model.

The paper is organized as follows. Section II briefly summarizes the authors' previous work on real-time price forecasting and concludes the problem left to be solved. Section III introduces the random forest. In section IV a detailed explanation of the proposed framework will be given. Section V gives the testing of the proposed model in NYISO and analyzes the advantages of the model. Section VI summarizes this paper's contribution and suggests the direction of future work.

II. ARMA AND ANN

A. Conducting an ARMA forecasting method

The notation AR (p) refers to the autoregressive model of order p. The AR (p) model is written as (1) where $\varphi_1,\ldots,\varphi_p$ are parameters, c is a constant, and the random variable ε_t is white noise.

The notation MA (q) refers to the moving average model of order q, it can be written as (2) where the θ_1, ..., θ_q are the parameters of the model, μ is the expectation of X_t, and the $\varepsilon_t,\varepsilon_{t-1},\ldots$are again, white noise error terms. [8]

The notation ARMA (p, q) refers to the model with p autoregressive terms and q moving-average terms. (3) contains the AR (p) and MA (q) models.

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t \tag{1}$$

$$X_t = \mu + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \tag{2}$$

$$X_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \tag{3}$$

An ARMA (144, 30) model is proposed here. Previous 30 days price historical data is required to train the model.

B. Conducting an ANN forecasting method

An artificial neural network is an interconnected group of nodes, akin to the vast network of neurons in a brain. [9] We propose two types of ANNs, one for weekend day (Saturday and Sunday) real-time price forecasting and the other one for weekday (Monday to Friday) real-time price forecasting. Three hidden layers are set in these two categories of ANNs with 8 neurons in the first hidden layer, 6 in the second layer, and 4 in the third layer. The inputs vectors are real-time price three hours ago, real-time price one day ago, real-time price one week ago, real-time price one month ago, current load, and day information (Monday to Sunday)

The output of the ANN is the current real-time electricity price. 8 previous similar days are need for training and 1 similar day for validation.

III. RANDOM FOREST

A. Introduction to Random Forest

The predecessor of Random Forest (RF), Classification and Regression Tree (CART), was proposed by L. Breman in 1984. Breiman introduced another essential technique for RF, called Bagging, in 1996. [10] RF is an ensemble learning method for classification. It is based on two techniques, the CART and the Bagging. The CART is a tree-structured classification model that maps observations about an item to conclusions about the item's class. A simple illustration of CART can be found in Fig. 1, where we can see that CART makes a decision in each node based on a split of a variable and makes its way down till reaching a leave node. Fig. 1 also provides a hint to the CART growing procedure: iteratively splitting each node into 2 sub-nodes by finding a best split variable along with a best split value till reaching minimum node size. CART's advantage is that it can be fitted into data perfectly well. However, when conducting prediction, CART's accuracy isn't that good. In other words, CART has low bias but suffers from high variance. [11]

To solve this problem, RF extends CART by introducing Bagging method. This means that 1) RF fits a multitude of CARTs into bootstrap sets resampled from the origin training set; 2) RF predicts through the mode of the predictions generated by the fitted CARTs. The introduction of Bagging will reduce the variance of CART while keeping the bias low. Moreover, RF adopts a trick called randomized node optimization to further reduce the CART variance. All above modifications to CART made by RF avoids the disadvantages of CART and proves to achieve very nice performance.

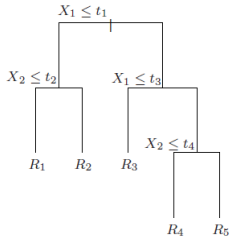

Fig. 1. A simple illustration of CART.

B. Random forest growing and predicting

The Breiman's Algorithm for constructing and predicting of RF is presented below. In brief, the procedure is to construct a set of CARTs fitted to bootstrap sampled datasets. Note in step 1-b-i and 1-b-ii, instead of searching for the best split variable among all p variables, RF limits the candidate best split variables to m randomly chosen variables. This is the randomized node optimization we mentioned above as a trick to reduce the CART variance.

Growing Stage:

Input:

(a)Training Data: N p-dimension samples along with their class labels.

(b)Require Parameter B: Number of trees.

(c)Require Parameter m: Number of candidate split variables at each split;

(d)Require Parameter n_{min}: Minimum node size.

1. For $b = 1$ to B:

(a) Draw a bootstrap sample Z^* of size N from the training data.

(b) Grow a random-forest tree T_b to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size n_{min} is reached.

i. Select m variables at random from the p variables.

ii. Pick the best variable/split-point among the m.

iii. Split the node into two daughter nodes.

2. Output the ensemble of trees $\{T_b\}_1^B$

To make a prediction at a new point *x*:

Let $\hat{Y}_b(x)$ be the prediction of the b_{th} random-forest tree. Then $\hat{Y}_{rf}^B(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{Y}_b(x)$

C. Statistical merits of random forest

In addition to the low bias & low variance, RF has several other desirable features, summarized below:

1) RF requires only 3 parameters. Following recommended values they are very easy to tune.
2) RF can generate an out-of-bag error, a nice estimate of the generalization error, in its growing procedure, while other models generally require multiple training procedures like Cross Validation to generate such estimates.
3) RF can generate variable importance indices in its growing procedure and they turn out to be nice estimates of variable relevancies.
4) RF is robust against irrelevant features and outliers in training data.
5) Structured as a tree, RF is in nature easy to expand itself to fit more data by growing more 'branches'. This leads to the RF online learning algorithm and has made RF a nice adaptive machine learning model. [12]

IV. Proposed Adaptive Forecasting Framework

A. Preprocessing and Predictors Definition

The data are preprocessed before being passed to the random forest model. The preprocessing consists of two steps. Firstly, price spikes in the training data will be discarded and the vacant values will be interpolated by nearby price values. Secondly, the training data will be smoothed, ignoring random fluctuations.

The predictors for the random forest consists of past price values.

TABLE I
PREDICTORS FOR THE RANDOM FOREST

Predictor	Description
$P(t-3)$ $P(t-24)$ $P(t-168)$ $P(t-720)$	Real-time price three hours ago, real-time price one day ago, real-time price one week ago, and real-time price one month ago
$L(t)$	Current load
$W(t)$	Current Temperature
$D(t)$	Day indicator (Monday to Sunday)

B. Constructing Confidence Interval

In the random forest predicting procedure, the prediction is set as the mean of the predictions of the trees, i.e., the mean of $\ldots\hat{Y}_b(x)$ $(b = 1 \ldots B)$.

Consider a set of observations Y, and assume a CART T generates the observations. Denote the posterior distribution of the CART parameters θ by $\theta\,|Y$. Consider T to be one of the trees in the random forest model, then its parameters, $\widehat{\theta}$, is estimated based on a bootstrap set sampled from Y. According to [13], such estimated parameters can approximate the posterior distribution of the tree parameters $\theta\,|Y$. As such, the trees in the random forest approximate the posterior distribution of the tree parameters $\theta\,|Y$. Thus, the predictions of the trees, $\hat{Y}_b(x)$ $(b = 1 \ldots B)$, reflect the posterior distribution of the prediction by the random forest. They can be used to construct the confidence interval attached to the prediction. In detail, firstly, estimate the density and distribution of the prediction using kernel density method, as shown in (4) (5); secondly, according to the confidence level α, and the probability distribution in (5), get the corresponding confidence interval.

$$f_Y(u) = \frac{1}{B\lambda\sqrt{2\pi}}\sum_{i=1}^{N} e^{-\frac{1}{2}(u-\hat{Y}_i(x))^2/\lambda^2} \tag{4}$$

$$F_Y(u) = \int_{-\infty}^{u} f_Y(v)\,\mathrm{d}v \tag{5}$$

C. Online Updating with New Observations

This part of the framework aim to update the random forest with new observations, so that the random forest can adjust to the latest forecasting condition. An online learning algorithm is introduced to update the random forest with a new observation, as shown below [14].

Require: Sequential training example $< x, y >$
Require: The size of the forest: T
Require: The minimum number of samples: α
Require: The minimum gain: β
// For all trees

for t from 1 to T do
  $k \leftarrow$ Poisson(λ)
  // Update k times
  for u from 1 to k do
    j = findLeaf(x).
    updateNode(j; $< x, y >$).
    if $|\mathcal{R}_j| > \alpha$ and $\exists s \in \mathcal{S}: \Delta L(\mathcal{R}_j, s) > \beta$ then
      Find the best test: $s_j = \arg\max_{s \in \mathcal{S}} \Delta L(\mathcal{R}_j, s)$
      createLeftChild($\mathbf{p}_{jls}$)
      createRightChild($\mathbf{p}_{jrs}$)
      UpdateGiniIndex.
    end if
  end for
end for
Output the forest F.

The key idea of the algorithm is summarized as follows.

1) In each tree, the new observation is processed repeatedly by k times, where k is Poisson distributed. This practice aims to simulate the bootstrap sampling procedure in the batch mode random forest learning algorithm.
2) The observation is processed as follows. Firstly it is passed down from the root node to the leaf node which the observation belongs to. Then, a decision will be made on whether to split the leaf node into two child nodes or not. The decision is made based upon whether the size of the leaf node is big enough and whether the possible reduce in training error is big enough if the split is made. [15]

V. Proposed Model Validation

A. Dataset Discription

The price data used in simulation model is downloaded from New York Independent System Operator (NYISO) [8]. The historical price data varied every 5 minutes from Jun 2nd, Sunday, Jun 8th, Saturday, Jun 9th, Sunday, Jun 15th, Saturday, Jun 16th, Sunday, Jun 22nd, Saturday, Jun 23rd, Sunday, Jun 29th, Saturday in New York HUD VL area are used to train the model. And we predicted the real-time price on that spot on Jun 30th, 2013, Sunday.

B. Simulation Settings

The simulation is divided into several steps. Firstly, a random forest is trained using historical data. It is then run on the test set and the accuracy will be recorded and compared with other benchmark models. Secondly, the adaptability of the model will be studied. The random forest model will be tested on a dataset recorded in a different market, economic or climatic scenario, to investigate how the random forest model can adjust to a different forecasting scenario, under which the mapping rules in the origin model will likely fail. The whole simulation procedure and the information about all the dataset used are summarized in Table II. The model parameters used in the tests are given in Table III.

TABLE II
SIMULATION SUMMARIZATION

Simulation	Simulation Description	Dataset Used
A	Test the predictive power of the random forest model.	For training: Jun 2nd, Jun 8th, Jun 9th, Jun 15th, Jun 16th, Jun 22nd, Jun 23rd, Jun 29th, 2013 … For test: Jun 30th, 2013
B	Train and test the benchmark models (ANN).	DITTO

TABLE III
MODEL PARAMETERS

Model	Parameters
Random Forest	B = 500 (Number of trees) m = 3 (Number of candidate variables in each split) n_{min} = 5 (Minimum node size) α = 90% (Confidence level)
ANN	Three hidden layers with 8 neurons in the first hidden layer, 6 in the second layer, and 4 in the third layer.
Adaptive Random Forest	α = 30 (The minimum number of samples) β = 0.1 (The minimum gain)

C. Simulation Results and Analysis

The results of simulation A and B are shown in Fig. 2 and Fig. 3, respectively. Fig.4 shows the ARMA simulation. Their corresponding accuracy is presented in Table III. It is shown that the random forest outperforms the ANN. Furthermore, most points fall within the confidence interval yielded by the random forest. The confidence interval provides useful information on the uncertainty of the predictions.

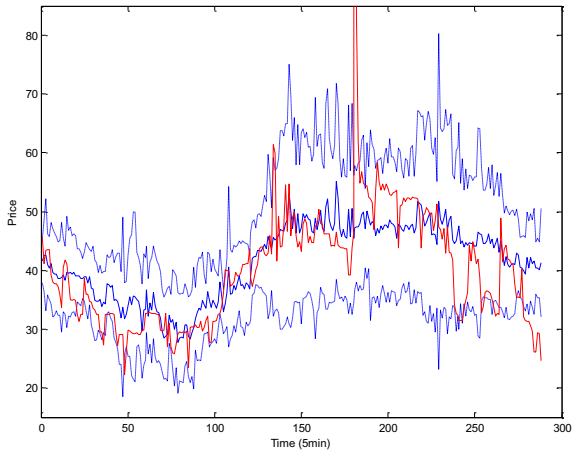

Fig.2. Predictions of the random forest model. The red solid line is true price curve and the blue solid line is the random forest predictions. The blue dotted lines represent the confidence level.

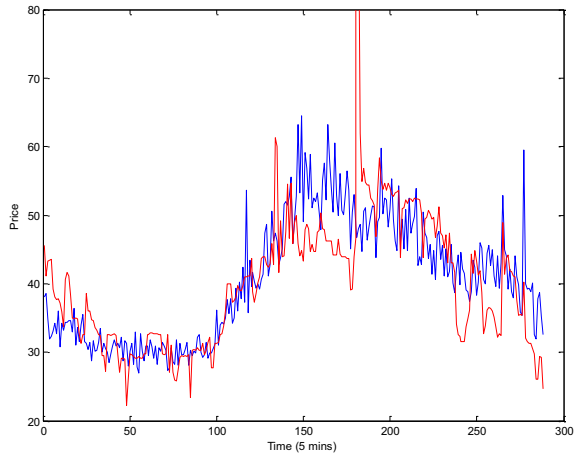

Fig.3. Predictions of the three-hour-ahead ANN. The red solid line is true price curve and the blue solid line is the random forest predictions.

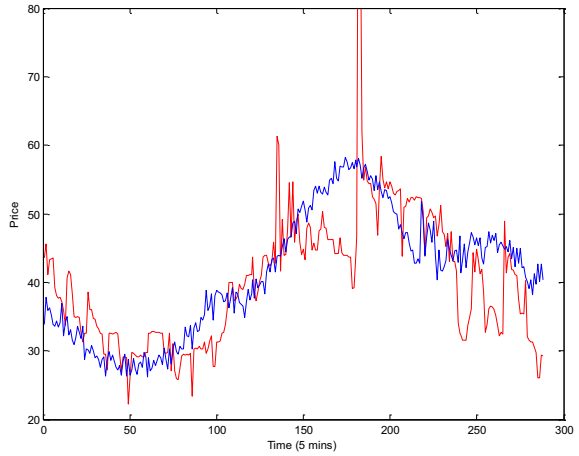

Fig.4. Predictions of the three-hour-ahead ARMA. The red solid line is true price curve and the blue solid line is the random forest predictions.

TABLE IV
FORECASTING RESULTS

Model	Random Forest	ANN	ARMA
MAPE	12.03%	12.83%	13.65%

VI. CONCLUSIONS AND FUTURE WORK

This paper proposes a random forest based adaptive model for real-time electricity price forecasting. The main contribution lies in the following two aspects. Firstly, the model provides a confidence interval associated with the prediction. Secondly, the model can adjust to the latest forecasting scenarios by updating itself with new observations. A case study has been presented to prove the validity of the proposed model.

We are still working to further study other factors that might affect the electricity price, such as real-time climatic and economic data. We are trying to build a predictor pool for the random forest. By utilizing the tree structure of the random forest, we are designing learning algorithms that can add new predictors to the predictor pool when online running. We also studying a copula based dependency analysis algorithm to discover the dependency structure of the predictors in the predictor pool and assess their importance for the prediction.

VII. REFERENCES

[1] Shahidehpour, Yamin and Zuyi Li. “Market operation in electric power systems” New York: John Wiley & Sons, Inc, 2002. Google books. Web. 20 Dec. 2012

[2] A.ott. “Experience with PJM market operation, system design, and implementation.” IEEE Transactions on power systems, vol. 18, no. 2, May 2003.

[3] Shahidehpour, Yamin and Zuyi Li. “Market operation in electric power systems” New York: John Wiley & Sons, Inc, 2002. Google books. Web. 20 Dec. 2012

[4] J. Contreras et al., “ARIMA models to predict next-day electricity prices.” IEEE Transactions on power systems, vol. 18, no. 3, Aug 2003.

[5] R. Garcia et al., “A GARCH forecasting model to predict day-ahead electricity prices.” IEEE Transactions on power systems, vol. 20, no. 2, May 2005

[6] N. Amjady . “Day-Ahead price forecasting of electricity markets by a new fuzzy neural network.” IEEE Transactions on power systems, vol. 21, no. 2, May 2006.

[7] S. Fan, C. Mao and L. Chen, “Next-Day electricity-price forecasting using a hybrid network.” IET gener. transm. distrib., vol. 1, no. 1, Jan 2007.

[8] P. Mandal et al., “An effort to optimize similar days parameters for ANN-based electricity price forecasting.” IEEE transactions on industry applications, vol. 45, no. 5, Sep 2009.

[9] Historical Real-Time Price data, Jul.2013 [Online]. Available: http://www.nyiso.com.

[10] Mills, Terence C. (1990) Times Series Techniques for Economists. Cambridge University Press.

[11] Minsky, M.; S. Papert (1969). An Introduction to Computational Geometry. MIT Press.

[12] Ho, Tin Kam (1998). "The Random Subspace Method for Constructing Decision Forests". IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8): 832–844.

[13] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning: Prediction, Inference and Data Mining, Second Edition, Springer Verlag, 2009.

[14] A. Saffari, C. Leistner, J. Santner, M. Godec and H. Bischof, “On-line random forests,” In Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, pp. 1393-1400, 2009.

[15] Breiman, Leo (2001). "Random Forests". Machine Learning 45 (1): 5–32. Tolosi L, Lengauer T (2011). "Classification with correlated features: unreliability of feature ranking and solutions.". Bioinformatics.