Day-Ahead Price Forecasting in Great Britain's BETTA Electricity Market

Daniel McGlynn, Sonya Coleman, Dermot Kerr, Catherine McHugh
University of Ulster
Magee Campus
N. Ireland
UK
{d.mcglynn; sa.coleman; d.kerr; mchugh-c24}@ulster.ac.uk

Abstract. The characteristics of commodities such as electricity, natural gas and oil mean that standard statistics-based pricing and prediction models that are typically applied in financial markets cannot readily be transferred and used as energy pricing models. Therefore, we investigate the use of computational intelligence-based approaches for electricity price forecasting. This paper compares two models for day-ahead electricity price forecasting, an AdaBoosted ensemble of the Extra-Trees algorithm and a Generalized Regression Neural Network (GRNN). In this work both forecasting models were applied to the national electricity market of Great Britain, the British Energy and Electricity Trading Arrangements (BETTA). The models were evaluated using the mean absolute percentage error (MAPE) statistic and the results show that the GRNN yielded a comparable forecasting error to the AdaBoosted algorithm with a significantly faster computation time.

Keywords: Extra-Trees, Generalized Regression Neural Networks (GRNN), British Energy and Electricity Trading Arrangements (BETTA), Machine Learning, Price Forecasting

I. INTRODUCTION

In most commodity markets the effect of production and supply on prices is dampened by surplus storage. Electricity cannot be stored in large quantities practically which is an intrinsic source of volatility in electricity price. Since the beginning of de-regulation of electricity markets, electricity price forecasting has become one of the main endeavors for researchers and participants in energy markets.

A good price forecasting tool in deregulated markets should be able to capture the uncertainty associated with such prices. Some of the key uncertainties are fuel prices, future addition of generation and transmission capacity, regulatory structure and rules, future demand growth, plant operations and climate change. These factors impact electricity price, and some factors are more important than others. Price forecasting tools are essential for all market participants for their survival under a deregulated environment.

The BETTA market consists of two ex-ante markets (day-ahead and intraday) and Balancing mechanism markets. It is important for participants of the BETTA market to establish a physical position in the ex-ante markets to reduce their exposure in the balancing mechanism where the price of energy is more volatile. Participants can further reduce their risk to price volatility by adopting arbitrage and hedging strategies prior to entering the ex-ante markets and an accurate price prediction is key to adopting such strategies.

Anbazhagan and Kumarappan [1] forecasted day-ahead electricity prices in the national electricity market of Singapore. They compared results from multi-layer neural network (MLNN) with levenberg-marqvardt (LM) algorithm, generalised regression neural network (GRNN) and cascade-forward neural network (CFNN). Prediction results corresponding to the market of Singapore for four weeks of the year 2006 was reported and yielded an average weekly MAPE of 11.78% for MLNN with LM, 11.94% for GRNN and 11.22% for CFNN. The inputs of their ANN models were the historical price and system demand. They selected lag prices based on a correlation analysis.

Mei et al. [2] proposed a real-time price forecasting through random forest. The model could adjust to the latest forecasting scenarios by updating itself with new observations. They compared the random forest results to an Autoregressive moving average (ARMA) model and an artificial neural network (ANN). The random forest could be updated with new observations. The MAPE of random forest was 12.03%, 12.83% for ANN and 13.65% for ARMA. P. Mandal et al. [3] proposed an improved ANN electricity price forecasting method in which a sensitivity analysis of similar day parameters was added to increase the model's accuracy. The improved model was tested in the Pennsylvania – New Jersey – Maryland (PJM) market and the MAPE was 11%.

The current BETTA market in Great Britain began trading electricity in April 2005. Each trading day in the day-ahead market consists of twenty-four hourly settlement periods. For each settlement period suppliers assess in advance what their demand is likely to be. They then contract with generators for that volume of electricity. Contracts can be struck up to the start of the settlement period. Every hour generation companies offer to sell electricity into the wholesale electricity market. All sales and purchases of electricity through the wholesale market are settled through the Nominated electricity market operator's (NEMO) trading platform. The BETTA market has two NEMOs Nordpool and EpexSpot.

In Great Britain's electricity market, the price of electricity in the day-ahead market varies every hour. Generators and suppliers and consumers require some expectation of future electricity prices to aid them in optimizing their business strategies. Given the importance of electricity price forecasting this paper focuses the use of two electricity price-forecasting techniques and day-ahead electricity prices from Great Britain's electricity transmission system. In Section II, we will introduce the data that are used in the computational intelligence techniques and describe how auto-correlation is applied to determine the best lag terms. We also outline the two computational intelligence approaches used, the Adaboosted algorithm and a Generalized Regression Neural Network (GRNN). The electricity price dataset was prepared using daily trading reports from BETTA. Section III presents price forecasting results of an AdaBoost ensemble of extremely randomised trees and the generalised regression neural network (GRNN) from the BETTA market. Section IV summarises this paper's findings and suggests the direction of future work.

II. METHODOLODY

This section describes the process of determining appropriate input lags (historical values) for the computational intelligence algorithms using autocorrelation and describes the machine learning algorithms used for day-ahead forecasting in Great Britain's electricity market.

1.Autocorrelation analysis

In order to perform the research reported in this paper, the electricity price data was obtained from Nordpool's day-ahead market reports [4]. The data set consisted of day-ahead market price for each hour. The natural gas price was obtained from National Grid's data item explorer [5], and system demand, solar generation, and wind generation was obtained from reports from national grid [6]. The most effective lags (price of the previous settlement periods) are selected by autocorrelation analysis which determines the similarity between observations as a function of the time lag between them.

A plot of the autocorrelation function (ACF) for the day-ahead electricity prices in the BETTA market is shown in Fig.1. The peaks indicate the time lags where historical prices are most highly correlated to the day-ahead market price. The dashed horizontal lines indicate a 95% confidence interval. The peaks of the ACF that are above this line indicate a strong positive correlation between these lag prices and the day-ahead market price. Therefore, the appropriate lags determined from the autocorrelation analysis are:

$$l \in L = \{1, 2, 3, 23, 24, 25, 47, 48, 49, 71, 72, 73, 95, 96, 97, 119, 120, 121, 143, 144, 145, 167, 168, 169, 191, 192, 193, 215, 216, 217, 239, 240, 241\}.$$

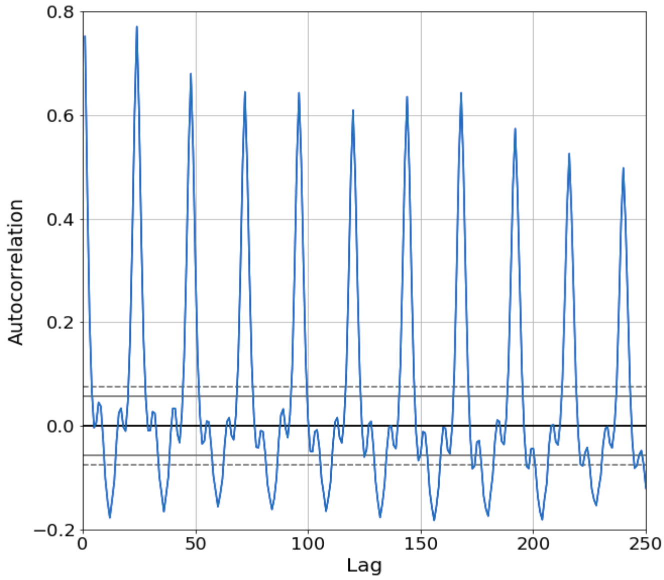

Fig 1. The autocorrelation function (ACF) for day-ahead electricity prices in BETTA 2017.

2. AbaBoost Ensemble of Extra-Trees.

In the first stage of the study, the AdaBoost algorithm [7] was used for electricity price forecasting. AdaBoost (also known as adaptive boosting) is a meta-algorithm that can be used with other types of algorithms to improve performance. The algorithm begins by fitting a regressor on the original dataset and then fits additional copies of the regressor repeatedly on the same dataset except that the weights of instances are adjusted according to the error on the current prediction. These additional copies focus on more difficult cases. In this study the dataset was fitted to a sequence of 50 models using a base learner.

The hyperparameters of the AdaBoost algorithm were tuned using the Randomized Grid Search method in Python's machine learning module Scikit Learn. A fixed number of parameter settings is sampled from specified distributions of hyperparameter values. A model is constructed and evaluated for each combination of parameters chosen. Tuning was performed five times and the set of parameters that resulted in the smallest error was subsequently used to forecast using the unseen test dataset. The hyperparameters that were tuned are shown in Table 1.

Table 1. AdaBoost algorithm parameters that were tuned using Random grid search for each settlement period hour.

Model	Parameters
AdaBoost	Base Estimator = Extra-Trees N = Number of estimators Learning rate: shrinks the contribution of each regressor Loss function: to use when updating the weights after each boosting iteration. It can be linear, square or exponential.

In this work AdaBoost was used with the extra-trees algorithm as the base learner. The Extra-Trees (Extremely randomized trees) was first proposed in 2006 [8]. It is a decision tree-based ensemble method with an adaptation of the random forest algorithm. A random forest is constructed from many decision trees at training time. The bootstrapping selection method is employed where several samples are randomly selected from the dataset with replacements and a decision tree is grown from these samples. This is repeated for several decision trees. Predictions for unseen samples are made by averaging the predictions from all the individual regression trees and the output of the Random Forests is combined into a weighted sum that represents the final output of the boosted regressor. During training Random Forest algorithm uses an objective function to determine how to splits nodes at a cut-point.

The Extra-Trees algorithm differs from the Random Forest algorithm in that it completely drops the bootstrapping of samples i.e. it grows decision trees from all the samples and it splits tree nodes at a completely random cut point. The usage of the full learning sample instead of bootstrap replicas is motivated to reduce bias. The randomization of the cut point reduces correlation between trees and so reduces variance and hence increases the generalization of the model. The hyperparameters of the Extra-Trees algorithm used in this study and their values are listed in Table 2.

Table 2. Extremely randomized trees and GRNN model parameters

Model	Parameters
Extra Trees	N = 50 (Number of trees) n_{max} = 10 (maximum number of leaf nodes) d_{max} = 25 (maximum depth of tree) $impurity_{min}$ = 0.1 (percentage decrease in node impurity after a split) $leafsample_{min}$ = 15 (minimum samples at a leaf node)
GRNN	σ (smoothing parameter)

3. Generalized Regression Neural Network

In this study, a GRNN was also used to forecast day-ahead electricity prices in the BETTA market for comparative purposes. The GRNN was devised by Specht [9] and the structure of the GRNN is a multilayer structure consisting of an input layer, radial basis layer, regression layer and an output layer. The structure of the GRNN is shown in Figure 2. All hidden units simultaneously receive an input vector. GRNNs are sensitive to situations where one input feature is significantly larger than the others. Therefore, as a preprocessing step, all input values were normalized so that all input values had similar scales.

The GRNN input features (historical prices, system demand, solar generation, wind generation, and gas price) were linearly normalized in the range [-1,1]. The outputs from the GRNN were then de-normalized before being presented in performance evaluation. The GRNN was implemented using the Python Artificial Neural Networks library NeuPy [10] and additional information about this GRNN can be found in the NeuPy documentation [11].

The smoothing parameter, σ, is required by the GRNN and must be specified. Specht's holdout method for selecting an appropriate value of the smoothing parameter was employed. In practice the holdout method is as follows. For a value of σ, one sample is removed before training the GRNN, then the trained GRNN is used to predict the sample that was removed. This is repeated for each sample and the root mean squared error (RMSE) is calculated between the predicted and actual values. The holdout method tested values of σ between 0.2 and 20 in incremental steps of 0.2. The value of σ that gives the smallest error is then used to train the GRNN.

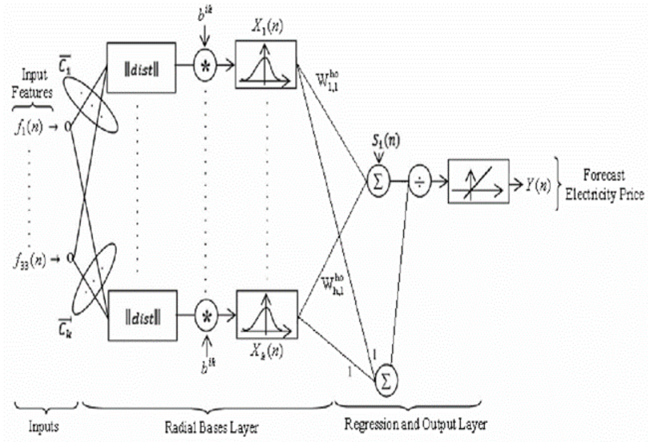

Fig. 2. Implementation of the generalised regression network for electricity price forecasting.

III. Results

In these experiments, data from the national electricity market of Great Britain, the British Energy and Electricity Trading Arrangements (BETTA) were used. All the samples in the dataset have thirty-seven features. Namely, these features are thirty-three historical day-ahead market prices, system demand, wind generation, solar generation and gas price. The input data set consisted of a sample for each hour from 2016 and 2017 which resulted in 15336 samples. These samples were filtered by settlement period i.e. samples from each settlement period (1-24) were separated and grouped together. The resulting dataset contained samples belonging to the same settlement period in each day and a model was trained and tested on these data. A forecasting model was trained for each of the settlement periods (1-24) for 639 days and it was tested on 92 days of unseen data from the months of October, November and December 2017.

An Adaboosted ensemble of the Extra-Trees algorithm and the GRNN were both trained using historical data and tested using the test set. The behavior of the trained algorithms was initially plotted for qualitative. The actual and forecasted prices using Adaboosted Extra-Trees for the months of October, November and December 2017 is shown in Figure 3.

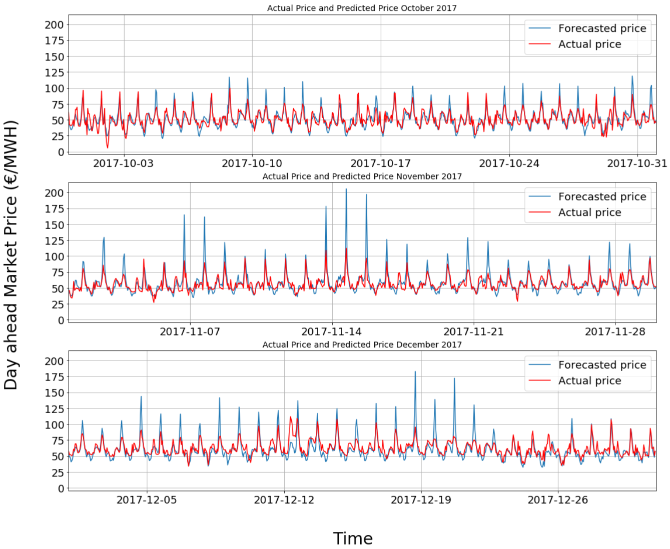

Fig 3. Day-ahead market price for Great Britain's electricity market: Actual prices (red), AdaBoosted Extra-Tress price forecasts (blue) in € per MWh.

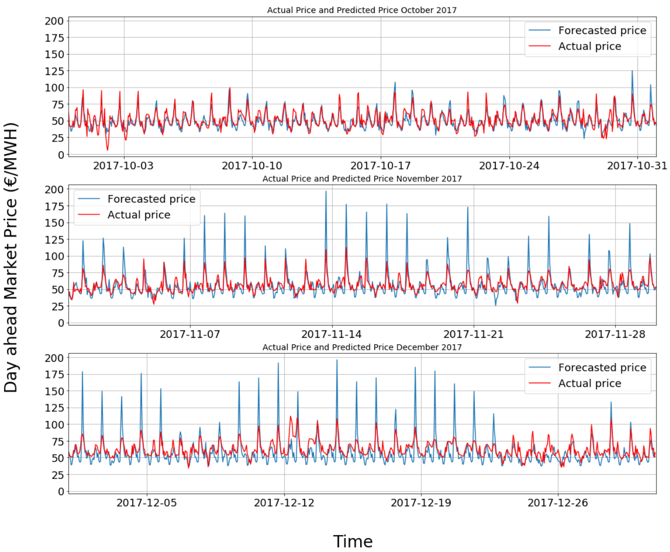

Fig 4. Day-ahead market price for Great Britain's electricity market for October, November and December 2017. Actual prices (red), GRNN price forecasts (blue) in € per MWh.

The actual and forecasted prices using GRNN for the months of October, November and December 2017 is shown in Figure 4. Each figure shows the actual day-ahead price in red and the forecasted day-ahead price in blue. In Figure 3, we can see that the AbaBoost algorithm generally follows the trend of the original data for each month, however is fails to ever reach the peaks in the data even though they are generally periodic. Similarly, in Figure 4, we can see that the GRNN algorithm generally follows the trend of the original data for each month, but it is clear in Figure 4(a) during October the GRNN is capable of modelling the data peaks although not quite recently the peaks in other months.

Quantitative evaluation is conducted by comparing the network predicted output value with the expected value via an evaluation statistic, Mean Absolute Percentage Error (MAPE). MAPE measures the deviation between the actual and forecasted price and is defined as:

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{A_i - F_i}{A_i}\right| \times 100 \qquad (1)$$

where A_i and are the actual and forecasted electricity price of the i^{th} settlement period respectively and N is the number of forecasted settlement periods. The simulations were carried out using an Intel Core i7-7500U CPU @ 2.7GHz and 8GB RAM and the results obtained are presented in Table 3. The results in Table 3 demonstrate that the GRNN has the lowest percentage error and standard deviation with a significant reduced training and test time in comparison to the AdaBoosted Extra-Trees. Therefore, the GRNN is the best approach from these two algorithms for energy price prediction in a real-time environment.

Table 3. Comparative MAPE results between Adaboosted extra-trees and a GRNN

Model	Adaboosted Extra-Trees	GRNN
Average MAPE %	11.59	13.99
Standard Deviation	4.58	6.88
Training time (s)	53.01	0.0019
Testing time (s)	1.37	0.23

IV. Conclusions

This paper presented two machine learning algorithms for day-ahead market price forecasting in Great Britain's electricity market. Autocorrelation was used to identify the most important historical lag prices that impacted the day-ahead market price. Historical lag prices together with Transmission system demand, gas price, wind generation and solar generation were used as input parameters to both algorithms. The previous 639 days were used for training each model and the next 92 days were forecasted. In the first approach, an AdaBoosted ensemble of Extra-Trees algorithm was used to forecast price. Then a GRNN with a smoothness parameter, selected using the holdout method, was trained and tested on the same price dataset. Prediction results corresponding to the market price for the 92 days of unseen data for 2017 are reported and show that the Adaboosted Extra-Trees gave a MAPE of 11.59% and the GRNN gave a MAPE of 13.99%. The GRNN had smaller forecasting error to the Adaboosted Extra-Trees and less computation time.

Further work is required to test additional boosting algorithms with neural network structures on BETTA day-ahead market price to further decrease error and maximise prediction accuracy for a real-time energy trading environment.

References

[1] S. Anbazhagan, N. Kumarappan. Day-Ahead price forecasting in Asia's first liberalised electricity market using artificial neural networks. Vol. 4. No. 4. June 2011, pp. 476-485.

[2] J. Mei, D. He, R. Harley, T. Habetler, G. Qu. "A random forest method for real-time price forecasting in New York electricity market" in Proc. Of the 2014 IEEE Power and Energy Society General Meeting, 2014, 29 October 2014, Washington, USA [Online]. Available: IEEE Xplore, http://www.ieee.org. [Accessed: 15 June 2018]

[3] P. Mandal, T. Senjyu and T. Funabashi, Neural networks approach to forecast several hour ahead electricity prices and loads in deregulated market, Energ. Convers. Manage. 47 (15-16) (2006) 2128-2142.

[4] Nordpool. (2018). Nord Pool [Online]. Available: https://www.nordpoolgroup.com

[5] National Grid (2018). Data Item Explorer [Online]. Available: http://mip-prod-web.azurewebsites.net/DataItemExplorer/Index

[6] National Grid. (2018). Electricity Transmission performance reports | National Grid UK [Online]. https://www.nationalgrid.com/uk/electricity/market-operations-and-data/transmission-performance-reports

[7] Y. Freund, R. E. Schapire. A Decision-Theoretic generalisation of on-line learning and an application to boosting. Journal of Computer and System Sciences. Vol. 55, 1997, pp. 119-139

[8] P. Guerts, D. Ernst, L. Wehenkel, Extremely randomized trees, Mach Learn (2006) 63:pp. 3–42, March 2006.

[9] D. F. Specht. A General Regression Neural Network, IEEE Transactions on Neural Networks. Vol. 2. No. 6. November 1991,pp. 568-576

[10] Y. Shevchuk. (2017). Home-NeuPy [Online]. Available: http://neupy.com

[11] Y. Shevchuk. (2017). Documentation-NeuPy [Online]. Available: http://neupy.com/pages/documentation.html