Received: 4 July 2019 | Revised: 23 July 2019 | Accepted: 10 August 2019

DOI: 10.1002/asmb.2518

SPECIAL ISSUE PAPER

WILEY

Deep learning for energy markets

Michael Polson[1] | **Vadim Sokolov**[2]

[1]Bates White, Washington, District of Columbia

[2]Systems Engineering and Operations Research, George Mason University, Fairfax, Virginia

Correspondence
Vadim Sokolov, Systems Engineering and Operations Research, George Mason University, Fairfax, VA.
Email: vsokolov@gmu.edu

Abstract

Deep Learning (DL) is combined with extreme value theory (EVT) to predict peak loads observed in energy grids. Forecasting energy loads and prices is challenging due to sharp peaks and troughs that arise due to supply and demand fluctuations from intraday system constraints. We propose a deep temporal extreme value model to capture these effects, which predicts the tail behavior of load spikes. Deep long-short-term memory architectures with rectified linear unit activation functions capture trends and temporal dependencies, while EVT captures highly volatile load spikes above a prespecified threshold. To illustrate our methodology, we develop forecasting models for hourly price and demand from the PJM interconnection. The goal is to show that DL-EVT outperforms traditional methods, both in- and out-of-sample, by capturing the observed nonlinearities in prices and demand spikes. Finally, we conclude with directions for future research.

KEYWORDS

deep learning, energy pricing, extreme value theory, locational marginal price, long-short-term memory, machine learning, peak prediction, PJM interconnection, rectified linear unit, smart grid

1 | INTRODUCTION

Deep learning (DL) is used to predict wholesale prices and then DL is combined with extreme value theory (EVT) to predict load on the energy grid. This is essential for the economic operation of resources as electricity grids operate without large amounts of storage so generation of energy (supply) within the system must always match the demand of energy (load). Our goal is to show that the combination of DL and EVT (DL-EVT) provides better predictions of tail behavior of loads versus traditional methods. The traditional approach to electricity price and demand prediction has been applying economic models based on firm behavior. More recently, data-driven analytics, using large datasets and machine learning techniques, has become popular as a tool to uncover price patterns.

Electricity price prediction is challenging due to a number of complex factors that impact intraday grid conditions, which create highly volatile price spikes. Our approach is to develop multilayer deep networks to capture nonlinearities and temporal patterns in energy prices and demand. EVT is used to provide an objective function that captures load spikes above a prespecified threshold. As supply must constantly adapt to meet changes in load, accurate predictions are essential for making informed short- and long-term generation decisions. Accurate anticipation of fluctuations in load, especially sharp fluctuations, would remove certain flexibility constraints allowing for efficient deployment of generation and grid resources.

The key to efficient electric grid management is understanding peak loads. At the day-to-day level, overestimating or underestimating peak loads can be costly. Overestimating peaks causes the system to have too much generation in

reserve, while underestimating peaks causes the system to call upon costly, but flexible, sources of energy to quickly meet the demand. Day-to-day prediction is complicated by the increase in renewable energy, whose pattern of generation does not always match the system's pattern of demand. This imbalance in supply and demand patterns adds to the volatility of the system's energy prices and complicates predictions.[1,2]

Forecasting supply and demand with standard models, however, fails to address the importance of peak prediction. Standard models aim to predict the mean level of the price and load and typically do not capture any extreme spikes in the data. Gaussian errors in the context of electricity markets neither captures the peaks nor the fat-tailed distribution seen in practice. Up until now, data-driven models were not flexible enough to capture the extreme nonlinearities in the price dynamics. Deep learners (DL) have recently been shown to have empirical success in large datasets forecasting problems with high-dimensional nonlinearities. In practice, long-short-term memory (LSTM) provides a framework for building spatiotemporal models.[3,4]

The rest of our article is organized as follows. Section 1.1 provides connections to previous work. Section 2 discusses traditional DL models. Section 4 describes the energy market for electricity and the PJM (a regional transmission organization [RTO]) interconnection. Section 3 provides our contribution and combines DL models with EVT. Section 2.1 describes the optimization algorithm used to estimate parameters of our DL-EVT model. Section 5 provides the algorithms used for load and price prediction for PJM. Finally, Section 6 concludes with directions for future research.

1.1 | Connection to previous work

Davison et al[5] develops a spatial statistical model for the extremes of a natural process. Peaks are modeled as an exceedance of a certain threshold. EVT provides the framework for the prediction of these exceedances, and it predicts the frequency of energy price exceeding a certain threshold.[6] The exceedance over a threshold allows to measure risk associated with high prices.[7] Incorporating EVT into DL allows us to capture the tail behavior of the price distribution. In particular, the likelihood functions, defined using the EVT framework, allow us to properly model price spikes. In the context of energy markets, capturing the spikes is a crucial as these are the central component of interest in the market. Furthermore, this approach provides an improvement over traditional DL approaches, which typically only focus on capturing the mean of a given distribution. Our work builds on that of Sigauke et al[8], which develops probabilistic EVT model and Shenoy et al[9] which uses generalized linear model, with EVT errors to model electricity demand.

Data-driven energy pricing models used to forecast hourly locational marginal prices (LMPs) have been studied previously.[10-13] LMP is simply a cost of electricity at different locations within a system. Those locations are special load and generation nodes on electric grid. For example, PJM has 11 000 such nodes (4700 load nodes and 6300 generating nodes) and LMPs is different from one node to another.

Hong et al[12] proposes neural networks to predict LMPs in the PJM Interconnection. Manda et al[14] uses neural networks to improve performance, and Catalao et al[10] and Kim et al[13] predict LMPs in Nord Pool, an electricity spot market located across Northern Europe. Wang et al[15] predicts prices at various hubs in the American Midwest with a stacked denoising autoencoder, exploiting local information to improve its predictive performance. Modeling wind generated-electricity is considered in Hering et al.[16] Our analysis extends the functional data analysis approach for electricity pricing developed by Liebl et al.[17]

Cottet et al[18] and Wilson et al[19] develop a random effect Bayesian framework to quantify uncertainty in wholesale electricity price projections. Jonsson et al[20] forecasts electricity prices while accounting for wind power prediction. Christensen et al[21] forecasts spikes in electricity prices. Heavy tails in electricity prices are modeled using multivariate skew *t*-distributions in Cottet et al.[18] Benth et al[22] addresses the non-Gaussian nature of price data using L'evy process. Dupuis[23] develops a detrended correlation approach to capture price dynamics within the New York section of the grid. Garcia et al[24] explains time-varying volatility in prices using GARCH effects for one-day price forecasting. Li et al[25] develops a fuzzy inference system to forecast prices in LMP spot markets. Subbayya et al[26] addresses the problem of model selection.

2 | DEEP LEARNING

DL models have two key advantages over other frameworks. First, DL models have the ability to analyze inputs of high dimensionality, for example, in the millions or greater. Second, DL models provide a very flexible function that can approximate complex relations between inputs and outputs. DL uses a composite of univariate semiaffine rather than traditional

additive functions. DL models can efficiently approximate high-dimensional functions $y = F(x)$. The following section describes the DL models used in this article.

Let y denote a low-dimensional output and $x = (x_1, \dots, x_p)$ a high-dimensional input. A deep network prediction, denoted by $\hat{y}(x)$, is defined by hierarchical layers

$$\begin{aligned} z_0 &= x,\ z_1 = a_1(W_1 z_0 + b_1), \dots, z_L = a_L(W_L z_{L-1} + b_L) \\ \hat{y}(x) &= W_{L+1} z_L + b_{L+1}, \end{aligned} \tag{1}$$

where $W_l \in R^{n_l \times n_{l-1}}$ is the weight matrix, $b_l \in R$ is the bias term, and n_l is the number of neurons in layer l. Here, we apply nonlinear activation function a_l elementwise to the activation vectors $W_l z_{l-1} + b_l$. Typical activation functions are rectified linear unit (ReLU) $a(u) = \max(u, 0)$ and sigmoid $a(u) = 1/(1 + e^{-u})$.

Specifically, the DL approach employs a series of hierarchical predictors comprising L nonlinear transformations applied to the input x. Each of the L transformations is referred to as a *layer*, where the original input is x, the output of the first transformation is the first layer, and so on, with the output $\hat{y}$ as the last layer. Layers 1 to L are called *hidden layers*. The number of layers, L, represents the *depth* of our routine. Linear regression is a particular case of a DL model with no hidden layers.

It is well known that shallow networks are universal approximators and thus can be used to identify any input-output relations. The first result in this direction was obtained by Kolmogorov,[27] who showed that any multivariate function can be exactly represented using operations of addition and superposition on univariate functions. Formally, for input $x = (x_1, \dots, x_n) \in [0, 1]^p$ defined inside an p-dimensional cube, there exist univariate continuous functions $a^{n,q}$, defined on $[0, 1]$ such that each continuous real function F defined on $[0, 1]^p$ is represented as

$$F(x_1, \dots, x_p) = \sum_{q=1}^{2p+1} a_q \left(\sum_{n=1}^{p} a^{n,q}(x_n) \right),$$

where each a_q is a real-value function. This representation is a generalization of earlier results.[28,29] Kolmogorov[28] showed that every continuous multivariate function can be represented in the form of a finite superposition of continuous functions of not more than three variables.

Recurrent neural network (RNN) is a specific type of architecture designed to analyze sequences, for example, time-series data. RNNs can capture electricity prices' time series properties. Recurrent layers capture long-term dependencies without much increase in the number of parameters. They learn temporal dynamics by mapping an input sequence to a hidden state sequence and outputs via a recurrent relationship.

A hidden state variable h_t is used by RNNs to represent information from past observations of the sequence and to predict current observation y_t. Given predictors x_t, the observed data y_t, and a hidden state h_t, then

$$\begin{aligned} y_t &= \sigma(W_1 h_t + b_z) \\ h_t &= \sigma(W_2[x_t, h_{t-1}] + b_h). \end{aligned}$$

Here $\sigma(x) = 1/(1 + e^{-x})$ is the sigmoid function applied componentwise and is used for calculating both the hidden vector h_t and the output vector y_t. The main difference between RNNs and feed-forward DL is the use of a hidden layer with an autoregressive component, here h_{t-1}. It leads to a network topology in which each layer represents a time step, and we index it by t to highlight its temporal nature. Figure 1 shows graphically the processing performed by an RNN network for each element of the sequential data (time series). A particular type of RNN, called LSTM, was proposed to address the issue of vanishing or exploding gradients in plain RNNs during training. A memory unit used in LSTM networks allows a network to learn which previous states can be forgotten.[30,31]

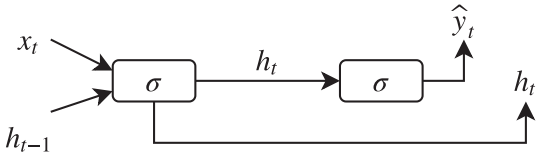


FIGURE 1 Hidden layer of a recurrent neural network

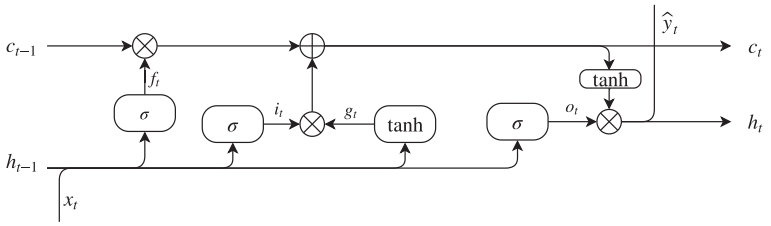


FIGURE 2 Hidden layer of a long-short-term memory (LSTM) model. Input (h_{t-1}, x_t) and state output (h_t, c_t)

The hidden state will be generated via another hidden cell state c_t that allows for long-term dependencies to be "remembered." Then, we generate

$$\text{Output: } h_t = o_t \star \tanh(c_t)$$

$$k_t = \tanh(W_c[h_{t-1}, x_t] + b_c)$$

$$c_t = f_t \star c_{t-1} + i_t \star k_t,$$

$$\text{State equations: } \begin{pmatrix} f_t \\ i_t \\ o_t \end{pmatrix} = \sigma(W[h_{t-1}, x_t] + b).$$

where $\star$ denotes the pointwise multiplication. Then, $f_t \star c_{t-1}$ introduces the long-range dependence. The states (i_t, f_t, o_t) are input, forget, and output states. Figure 2 shows the network architecture.

The key addition versus RNN is the cell state c_t, and the information is added or removed from the memory state via gates defined via the activation function $\sigma(x)$ and pointwise multiplication $\star$. The first gate $f_t \star c_{t-1}$, called the forget gate, allows to throw away some data from the previous cell state. The next gate $i_t \star k_t$, called the input gate, decides which values will be updated. Then, the new cell state c_t is a sum of the previous cell state c_{t-1} passed through the forgot gate plus selected components of the k_t vector, which is a filtered version of inputs (h_{t-1}, x_t). Thus, the vector c_t provides a mechanism for dropping irrelevant information from the past, and adding relevant information from the current time step. At the last output layer, the filtered version of of the previous hidden state and input vectors o_t is then combined with tanh applied to the cell state $o_t \star \tanh(c_t)$. The forget gate resolves the problem of vanishing gradient, which is the case when values of the gradient vector are close to zero. Stochastic gradient descent (SGD) optimization algorithm is straight forward to implement. See Section 2.1 for discussion.

Deep ReLU with LSTM cells have became popular architectures as they can capture long-range dependencies and nonlinearities. Their popularity stems from the fact that they can efficiently approximate highly multivariate functions with small number of neurons at each layer.[32-34]

2.1 | Stochastic gradient descent

Once the activation functions depth L and size $n_1, \ldots, n_L$ of the learner have been chosen, the parameters $\hat{W}$ and $\hat{b}$ are found by solving the following optimization problem

$$\text{minimize}_{W,b} \sum_{i \in C} l(y_i, F_{W,b}(x_i)) + \phi(W), \tag{2}$$

which is a penalized loss function, where $(y_i, x_i)_{i \in C}$ is training data of input-output pairs, and $\phi(W)$ is a regularization penalty on the network weights. Most architectures employ regularization techniques to prevent the model from overfitting training set data.[35] This improves the model's predictive performance on data outside of the training set. Normally, a regularization penalty allows to improve convergence rate of the optimization algorithms and to avoid overfitting. Dropout, the technique of removing input dimensions in x randomly with probability p, can also be used to further reduce the change of overfitting during the training process.[36]

A typical choice for regression problems is the Gaussian loss function $l(y_i, F_{W,b}(x_i)) = \|y_i - F_{W,b}(x_i)\|_2$, then we have a traditional least-squares problem[37] and $\phi(W) = \lambda ||W||_2$.

The common numerical approach to find the solution to this optimization problem (2) is SGD. It iteratively updates the current iterated by taking a step in the direction opposite to the gradient vector

$$(W, b)^+ = (W, b) - \eta \nabla \left[L(W, b) + \phi(W) \right].$$

SGD then uses backpropagation algorithm to calculate the gradient at each iteration. Backpropagation is an implementation of chain rule applied to a function defined by a neural network. One caveat of SGD is the complexity of the system to be optimized, resulting in slow convergence rates. As a result, DL methods rely heavily on large computational power.[38,39]

3 | DEEP LEARNING EXTREME VALUE THEORY

A traditional DL regression model uses least squared loss to estimate model parameters (weights and biases of each of the neural network layers). This model is not appropriate for quantifying large values of y (spikes) that are a rare but very crucial to the stable operations of electric grids. EVT approach allows to model the tail behavior of the distribution of electricity loads.

3.1 | Extreme value theory

We fit the generalized Pareto (GPar) distribution, parametrized by scale parameter σ and shape parameter ξ, using observations that exceed the threshold. Compared to the classical EVT that only models maximum values, each exceedance is associated with a specific time point and it allows us to incorporate covariates, for example, when parameters σ and ξ depend on input variable x.

Let each observation follow a common distribution $y_i \sim G(y_i) = \Pr(Y \leq y_i)$ and let $M_n = \max\{y_1, \dots, y_n\}$. The central result of the EVT is that regardless of the distribution G, the scaled value of M_n follows a limiting distribution K

$$\Pr\left\{ \frac{M_n - b_n}{a_n} \leq y \right\} = G^n(a_n y + b_n) \to K(y).$$

Here $a_n > 0$ and b_n are normalizing constants. Gnedenko[40] provided a rigorous mathematical proof of existence of this limiting distribution and characterized its functional form. Modeling the extreme values M_n limits the number of samples that can be used for estimation. For example, we can use monthly maximum loads but then we will have to discard most of the samples. Furthermore, traditional EVT does not naturally allow for covariates (predictor inputs). It makes this approach impractical. Smith[41] proposed to model values of y that exceed some fixed threshold value u. The distribution over the excess values has a limiting distribution as $u + y$ approaches the right-hand endpoint of the underlying distribution. Specifically, as $u + y$ approaches the right-hand side endpoint of distribution G, for some normalizing constant $c_n > 0$, we have

$$\Pr\{Y \leq c_u(u + y) | Y > u\} \to H(y),$$

where

$$H(y \,|\, \sigma, \xi) = 1 - \left(1 + \xi \frac{y - u}{\sigma}\right)_+^{-1/\xi}, \quad \xi \neq 0.$$

The distribution $H(y)$ is generalized Pareto distribution with density

$$h(y \,|\, \sigma, \xi) = \frac{1}{\sigma}\left(1 + \xi \frac{y - u}{\sigma}\right)^{-1/\xi - 1}, \quad 1 + \xi \frac{y - u}{\sigma} > 0, \quad \xi \neq 0.$$

Here (u, σ, ξ) are the location, scale, and shape parameters, $\sigma > 0$ and $z_+ = \max(z, 0)$. The exponential distribution is obtained by continuity as $\xi \to 0$, and we have

$$\lim_{\xi \to 0} h(y \,|\, \sigma, \xi) = \sigma \exp\left(-\sigma(y - u)\right).$$

Under this distribution, the mean value of y is $\sigma + u$.

3.2 | Deep learning EVT

Suppose that we have data denoted by $y(s_i, t_j)$ at spatial locations s_i, $1 \leq i \leq n$ and time t_j, $1 \leq j \leq T$. We build an EVT-DL input-output model for each location s_i and estimate it using the pairs $\{(y(s_i, t_k), x_k)\}_{k \in C}$, where $C = \{j | y(s_i, t_j) > u\}$ and $x_k = (y(s_i, t_k), y(s_i, t_{k-1}), \dots, y(s_i, t_{k-h}))$ are the recent observations of the output for the given location. We assume that observations follow generalized Pareto distribution with parameters being functions of the input variables

$$y(s_i, t_{k+\eta}) | x_k \sim \text{GPar}(\sigma(x_k), \xi(x_k)),$$

where η is the forecasting horizon. We model the functions $\sigma(x|W, b)$ and $\xi(x|W, b)$ using a DL model parametrized by weight matrices $W = (W_1, \dots, W_{L+1})$ and biases $b = (b_1, \dots, b_{L+1})$. Linear regression generalized Pareto model was developed in Davison et al[6] and Beitlant et al.[42] To complete our specification for exceedance sizes, we assume a functional form for $\sigma(x|W, b)$ that is a deep neural network. As shown in Equation (1), we introduce

$$(\xi(x), \sigma(x)) = F_{W,b}(x), \text{where } F = f_l \circ \dots \circ f_L, \; f_l(z) = \sigma(W_l z + b_l).$$

Here F is a deep learner constructed via superposition of semiaffine univariate functions, see Dixon et al,[3] Polson et al,[43] and Polson et al[44] for further discussion.

To estimate the weights and bias parameters of the DL model, we use the negative log-likelihood loss function. Under the assumption of Generalized Pareto distribution for our dependent variable, for a single observation, the negative log-likelihood is given by

$$l(y_i, F_{W,b}(x_i)) = \log \sigma(x_i) - (1/\xi(x_i) + 1) \log(1 + \sigma(x_i)\xi((x_i))(y_i - u)).$$

Then, the loss function for our DL model, which is the negative log-likelihood for a training data set, becomes

$$L(W, b) = \sum_{i \in C} l(y_i, F_{W,b}(x_i)).$$

The weights W and offsets b are learned by minimizing the loss function, using the SGD algorithm.

4 | ENERGY PRICES IN PJM INTERCONNECTION

The PJM Interconnection is an RTO, which exists to create a competitive wholesale electricity market, coordinating numerous wholesale electricity producers and consumers in all or parts of 13 states located in America's Mid-Atlantic and Great Lakes Regions as well as the District of Columbia.

PJM is divided into 20 transmission zones. Each zone is owned and operated by separate transmission owners who are responsible for designing and maintaining their portion of the system. Figure 3 shows PJM's load nodes and zone boundaries. Individual utilities within PJM plan their use of resources around peak loads. Predicting the strength and timing of these peaks is integral to improving both short- and long-term decision-making. Current methods used for short-term prediction focus on neural networks (weather channel, PJM).

PJM acts as a guarantor of system reliability and is responsible for preventing outages within the system. PJM operates the system at a cost-efficient level by coordinating generating plant operations, which are owned by various entities, to match the system's demand. Operating the system includes ensuring real-time demand is met, maintaining a reserve capacity of generation, and monitoring transmission lines to prevent overloaded lines, which could cause system failure.[45]

The PJM Interconnection contains over 11 000 nodes for which hourly day-ahead (DA) or real-time (RT) prices are reported. These nodes are specific generation or load locations, aggregates of various locations, regions, or points of interconnection with areas outside of PJM. Within the PJM Interconnection, nearly all wholesale electricity is bought or sold through bilateral contracts. The remainder is bought or sold on the two bid-based electricity markets PJM operates: DA and RT markets. In the DA market, market participants submit bids or offers to buy or sell energy to the scheduling operator (PJM). The operator uses the bid and offers to determine the DA LMP, which reflects the *expected* cost of energy, congestion, and transmission loss needed to provide electricity at a location given the *expected* system constraints.

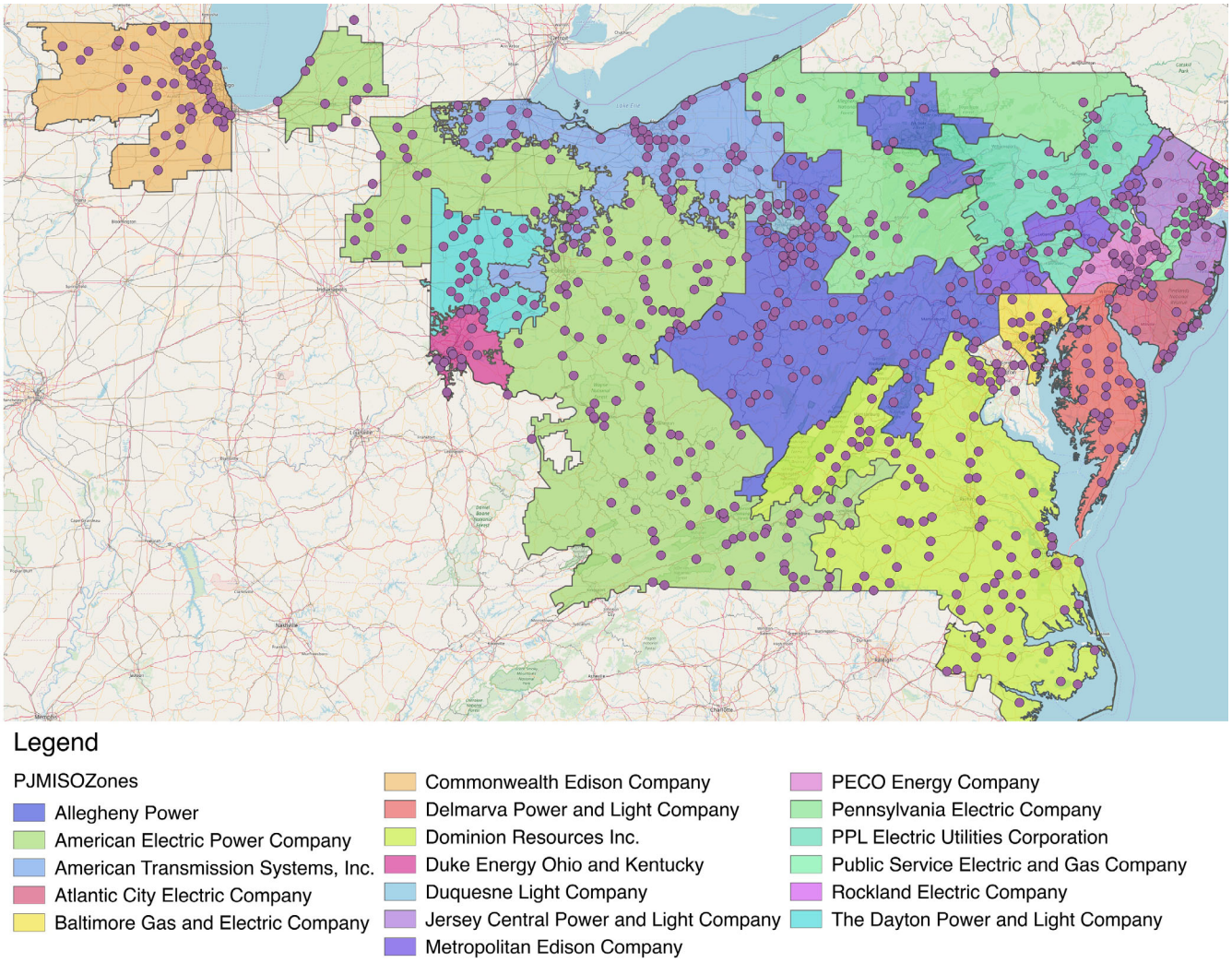


FIGURE 3 PJM zone boundaries and node locations [Color figure can be viewed at wileyonlinelibrary.com]

The RT market operates similarly but reflects the *actual* cost of providing electricity at a location given *actual* system constraints. Despite the smaller volume, the RT market plays a central role in determining the price of all futures contracts as the futures contracts' prices depends on the expected of the RT market prices. The DA market is a futures market that allows generators to agree to provide electricity for the upcoming day.

Generators can fulfill obligations to provide energy by either producing electricity or purchasing it on the RT market. Multiple factors, such as unexpected maintenance, may cause a generator to fulfill their obligation through purchases on the RT market rather than generation. These factors, or risks, cause significant volatility in RT markets compared to the DA market.[46]

Prices in the RT market are a function of the cost to produce electricity and system constrains, such as congestion in transmission lines. When these constraints are binding, prices differ across locations in the PJM Interconnection to reflect the relative ease of delivering energy to a noncongested location and the relative difficulty of delivering energy to a congested location. Therefore, each node (or location) has an associated LMP, which reflects the price of the marginal unit of electricity delivered to that specific location. LMPs are important price signals in the DA and RT market, which inform short-term decisions, as well as long-term investments and bilateral agreements.[45]

4.1 | Local marginal price data (LMP)

Locational marginal pricing is used to price energy on the PJM market in response to changes in supply and demand and the hardware's physical constraints. LMP accounts for the cost to produce the energy, the cost to transmit this energy

within PJM RTO, and the cost of energy lost due to resistance as the energy is transported across the system. LMP data is available at *www.pjm.com*.[47] Our study uses price data, which includes RT and DA hourly prices from 1 January 2017 to 31 December 2017. Load prices represent the cost of providing electricity at a given location. The price reflects the system's load (demand), generation, and limits of the transmission system. The system's constraints can affect locations asymmetrically, causing variations in price across different locations. Hub prices are a collection of these locational prices and reflect the uncongested price of electricity.

LMPs have three components: energy, congestion, and marginal loss. The energy component reflects the price of electricity, called system marginal price (SMP). SMP is calculated based on the current dispatch (supply) and load (demand). SMP is calculated for both the day ahead and RT markets. The congestion component is greater than zero whenever congestion occurs at a node. Constraints occur when delivery limitations prevent the use of least-cost generator, for example, a higher cost generator located closer to load must be used to meet the demand if transmission constraints are present. The congestion price is calculated using the shadow price, which is the value of the dual variable (price of violating a binding constrain) in the optimization problem that governs the grid. When none of the constraints are active, all the congestion prices are zero.

The marginal loss component reflects the cost of transmission and other losses at a given location. Losses are priced according to marginal loss factors, which are calculated at a bus and represent the percentage increase in system losses caused by a small increase in power injection or withdrawal.

5 | EMPIRICAL RESULTS

In this section, we begin with exploratory data analysis and then compare temporal neural network architecture with more traditional Fourier and ARIMA models to predict electricity prices. Furthermore, we demonstrate our DL-EVT approach to predict peak loads on the PJM interconnect.

5.1 | PJM price forecasting

We start with exploratory data analysis. First, we plot correlation matrix for prices at different zones to show spatial correlations of prices at different locations. Zone is a collection of several nodes bounded by a polygon. Figure 3 shows geographical boundaries of each of the zones. Figure 4 shows correlations between prices aggregated to zone level. The red-to-green color scheme was used, red indicates −1, green +1, and yellow indicates 0 (no correlation). The top part of the figure shows the color key. Each cell of the bottom part of the plot is the correlation between prices at two different zones. We used hourly observations from August 2017, a total of 651 observations at each zone to calculate the correlations. Figure 4 shows that there are strong spatial correlations among prices at different zones. Thus, prices at nodes will be correlated as well. Thus, we included prices at other locations as predictors for our models.

To show the nonlinear nature of relations between predictors and the price, we use several exploratory plots. We use data measured at a load node 48666 in Clifton, NJ. We used hourly data from first seven days of August 2017 (168 hourly observations).

Figure 5A shows hourly time series plots of price and load (demand). Load is measured in megawatt (MW) and price is measured in dollars per megawatt-hours (MWh). Figure 6B compares time series plots of price and temperature, measured in Celsius.

Figure 6 plots price observations against lagged values of price during the first seven days of August 2017. We used a 6-hour lag. Figures 5 and 6 show that relations between price, demand, and weather, and temporal relations in price data are nonlinear.

We build an individual model to predict price at each load point using observations of temperature, load, and price from the past 6 hours as predictors. Although Figure 4 shows that there is spatial correlation between observed prices at different locations, we found that by including price observations from other locations as predictors did not improve quality of forecast.

We show our forecasting model for predicting price at node 48666. Data from other nodes on the system can be modeled in the same way. Data from the same node were used for the exploratory plots above. First, we try traditional model for electricity prices, which uses Fourier series to describe the seasonal patterns and short-term time series dynamics

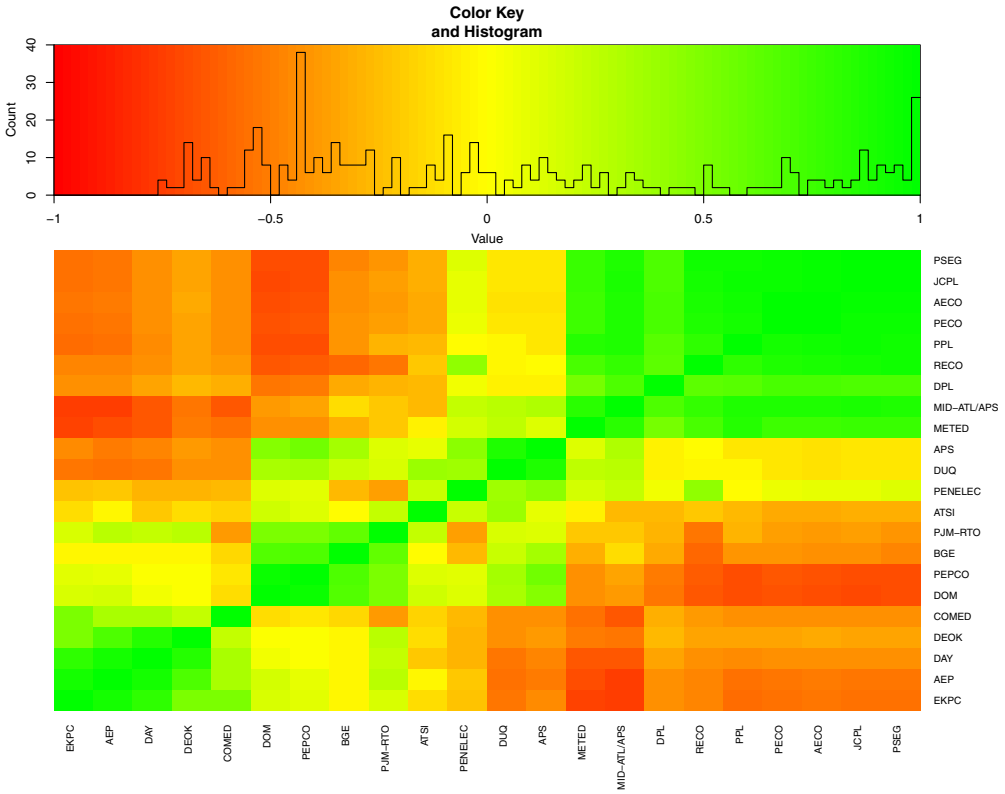
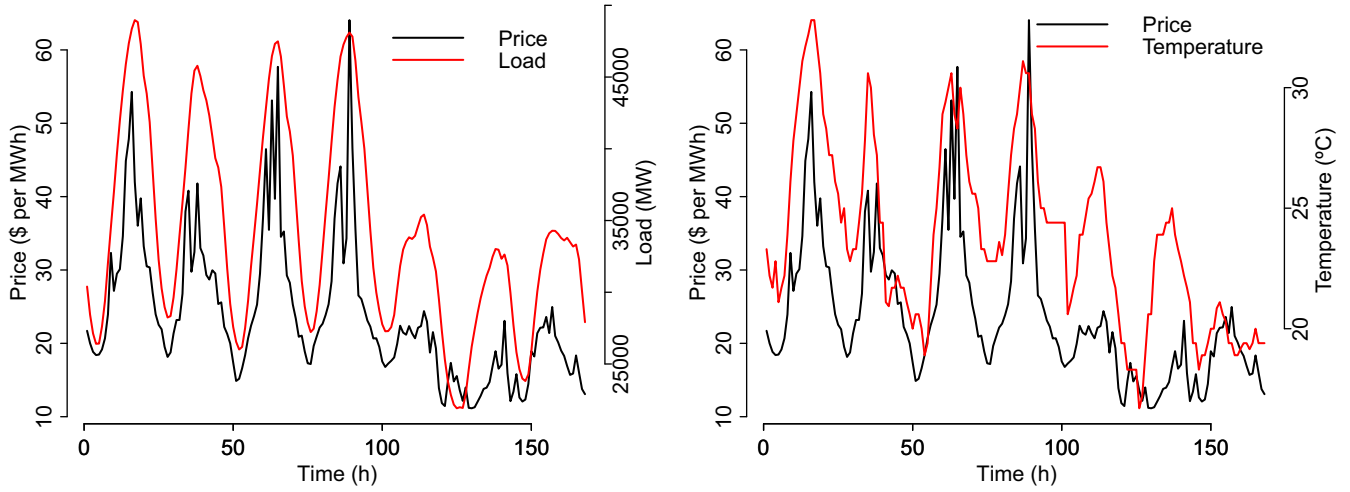


FIGURE 4 Correlation in marginal prices among zones. Each cell of the bottom plot indicates price correlation between a pair of zones. Red indicates −1, green +1, and yellow indicates 0 (no correlation). The top part of the figure shows the color key and histogram for correlation values, each bar is the count of occurrences of specific value [Color figure can be viewed at wileyonlinelibrary.com]



(A) Price vs Load (B) Price vs Temperature

FIGURE 5 Time series plots of price and load (A) and price and temperature (B). This plot shows relations between dependent variable price and independent variables load and temperature [Color figure can be viewed at wileyonlinelibrary.com]

modeled using ARIMA model. Here y_t (price at node 48666 at time t) is decomposed as a sum of a deterministic Fourier term $f(t)$, and regression term $\beta^T x_t$, and the stochastic component, N_t, leading to

$$y_t = \beta^T x_t + f(t) + N_t, \quad \text{where} \quad f(t) = \sum_{k=1}^{K} \left[\alpha_k \sin(2\pi kt/m) + \beta_k \cos(2\pi kt/m)\right], \tag{3}$$

where N_t is an ARIMA process. For predictors x_t, we use load, price, and temperature observations over the last 6 hours. Thus, we have 18 predictors. During prediction, we use naive forecasting schema for predictors and use values of the

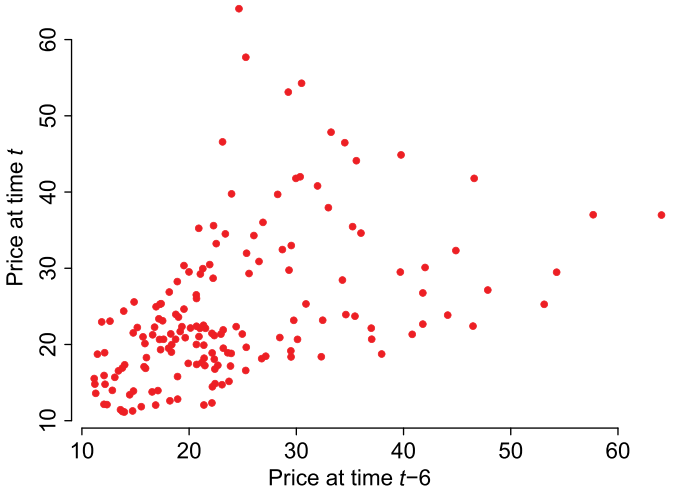


FIGURE 6 Temporal patterns in price data. Scatter plot of pairs od prices ($/MWh) at time t and $t-6$ [Color figure can be viewed at wileyonlinelibrary.com]

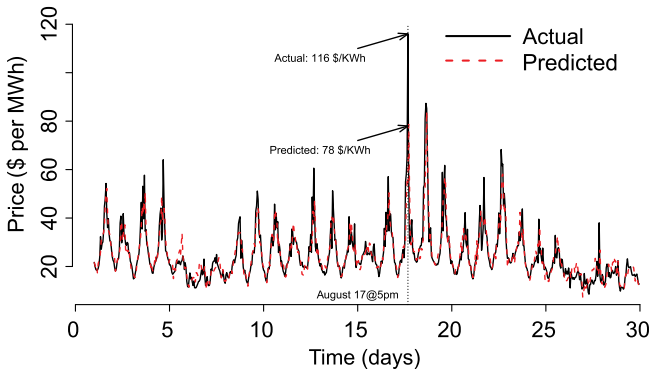


FIGURE 7 In-sample prediction by our Fourier-ARIMA model with predictors, described by Equation (3). Black solid line is the hourly price for the first 30 days of August 2017. Red dashed line is the predicted price for the same period [Color figure can be viewed at wileyonlinelibrary.com]

predictors from the past. For example, when we predict price for tomorrow at 5 PM, we assume that temperature and load will be the same as today at 5 PM. We develop a 2-day forecast. The time lag of 6 hours was chosen empirically. Using more than 6 recent hourly observations did not lead to any improvement in predictive power of our models.

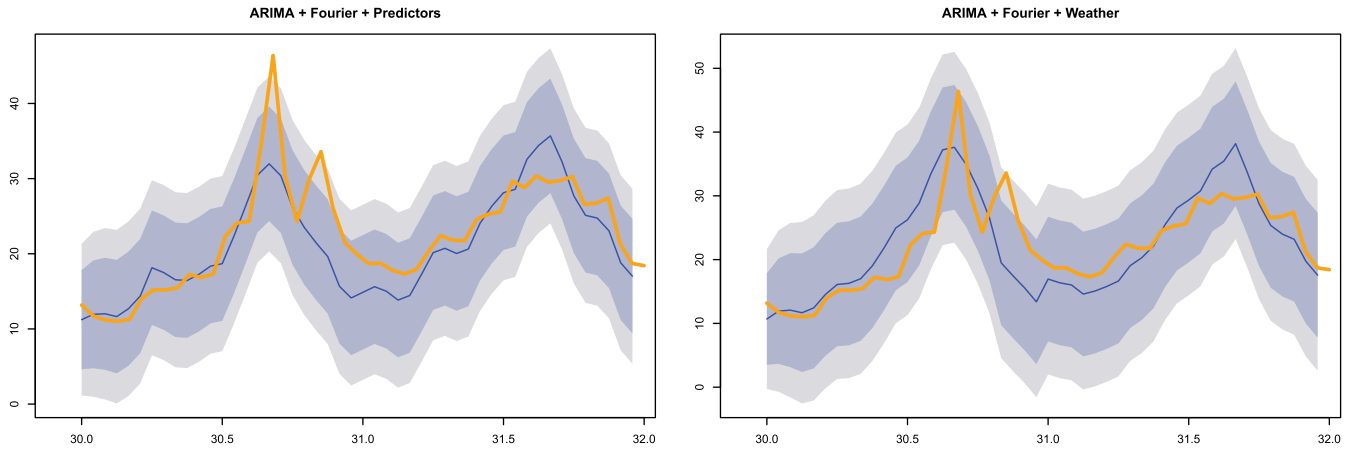
The number of terms K was chosen by minimizing the cross-validation error. Fourier term allows: (a) any length seasonality and (b) several seasonality periods. Smoothness of the seasonal term is governed by value K. The short-term dynamics is handled with an ARIMA error. The only real disadvantage (compared to a seasonal ARIMA model) is that the seasonality is assumed to be fixed—the pattern is not allowed to change over time. In practice, seasonality is usually remarkably constant so assumption generally holds except in applications with very long time series.

We use first 30 days of August 2017 to train our model. The in-sample fit of our Fourier-ARIMA model with predictors, described by Equation (3) is shown in Figure 9. This model captures the cyclical patterns in the prices but does not accurately capture the levels of the peak prices. The model predicts the time of the peaks but not the amplitude. For example, for August 17 at 5 PM (point marked by dashed vertical line on Figure 7) the actual price is 116 $ per MWh. Although our model does predict the time of the peak (5 PM), the predicted price is $78 per MWh. Thus, the value of the peak price is mispredicted.

Figure 8 shows the out-of-sample prediction for the next two days (August 31 and September 1 of 2017) of price for Fourier model with weather and demand predictors. Inclusion of predictors does not change the quality of forecasting peak prices. As we noted in our exploratory plots, demand is not a good predictor of a peak price.

We compare the Fourier model given by Equation 3 with temporal neural network (LSTM) model. Table 1 shows several out-of-sample fit metrics. Specifically, we show Mean Squared Error (MSE), Root Mean Square Error (RMSE), mean absolute error (MAE), and mean absolute percent error (MAPE).

While LSTM model shows an improved out-of-sample performance when compared to traditional ARIMA model with Fourier predictors, as shown in Figure 9, both the traditional ARIMA and LSTM neural network model are not capable to capture the peak in the price value at 5 PM of 31 August (hour 17 in Figure 9). Furthermore, the peak price lies outside of



(A) Demand + Weather (B) Only weather

FIGURE 8 Out-of-sample prediction of price from linear model with ARIMA$_{(2,0,0)}$ errors and Fourier predictors with $K = 5$. Yellow line is actual price (data) and the blue line is the forecast. Purple and grey areas are the 95% and 99% confidence intervals, respectively. *y*-axis is price in $ and *x*-axis is day from 1 August 2017. A, Demand + weather. B, Only weather [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Out-of-sample performance of DL and Fourier models

	MSE	RMSE	MAE	MAPE
Fourier + ARIMA	26.6	5.1	4	0.19
LSTM	16.8	4.1	2.4	0.09

Abbreviations: DL, deep learning; LSTM, long-short-term memory; MAE, mean absolute error; MAPE, mean absolute percent error; MSE, mean squared error; RMSE, root mean square error.

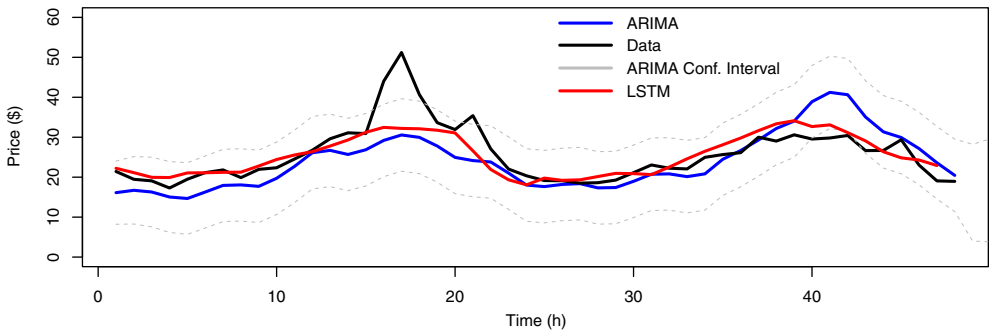
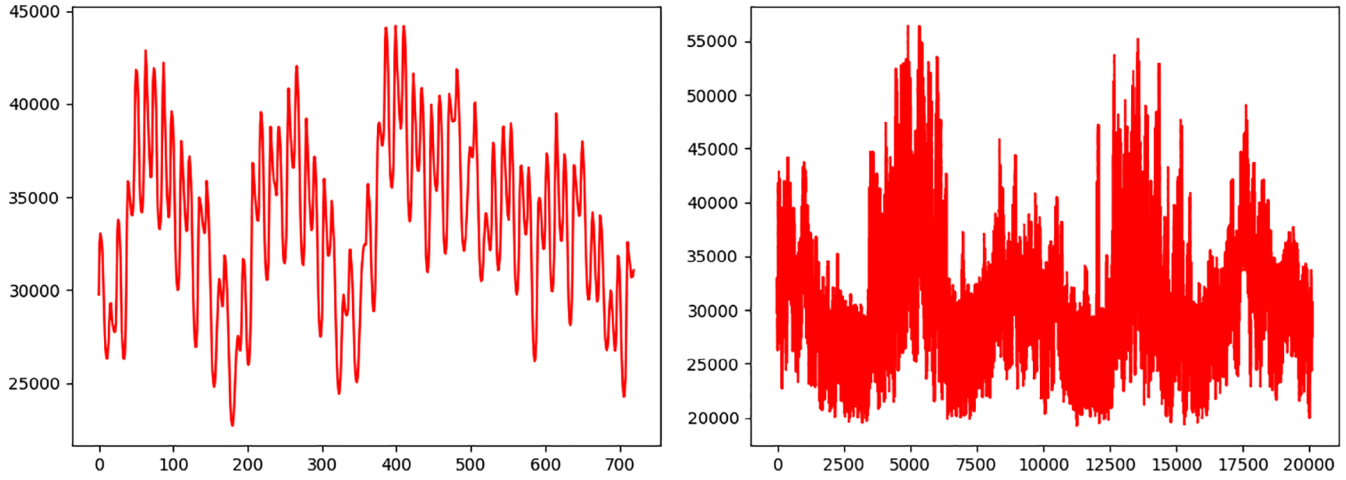


FIGURE 9 Comparison of Fourier and DL models. Black line is price in $/MWh for the period of 31 August-1 September 2017. Red line is price predicted by the long-short-term memory (LSTM) model, and blue line is the ARIMA model, which is given by Equation 3 Grey lines show confidence interval predicted by the ARIMA model [Color figure can be viewed at wileyonlinelibrary.com]

the 95% confidence interval of our ARIMA model. On the other hand, prediction of the peak values is of high importance. In the next section, we show how EVT combined with DL (DL-EVT) addresses this problem and captures the peak values of the demand time series.

5.2 | Demand forecasting

Electricity load forecasting is essential for designing operational strategies for electric grids. In presence of renewable energy sources, short-term forecasts are becoming increasingly important. Many decisions, such as dispatch scheduling



(A) Jan 1 - Feb 29, 2016 (B) Jan 1 2016 - Apr 17, 2018

FIGURE 10 Hourly electricity load on PJM interconnect in MW. The *x*-axis is the hour since 1 January 2016 and the *y*-axis is the load measured in MW. A, 1 January-29 February 2016. B, 1 January 2016-17 April 2018 [Color figure can be viewed at wileyonlinelibrary.com]

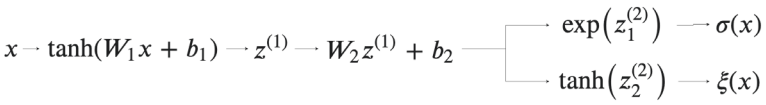


FIGURE 11 Neural network architecture used for deep learning extreme value theory (DL-EVT) model

and demand management strategies, are based on load forecasts.[48] One hour-ahead forecasts are a key input for transmission companies on a self-dispatching markets.[49] Hourly behavior of electricity load is known to be nonstationary.[50] Since there is not much of a change in meteorological variables, it is typical to use univariate time series data for short-term load forecasting.[51]

In this section, we analyze an hourly electricity load observations on the PJM interconnection. The data are available at https://www.dropbox.com/sh/1dczb673bx9kxzl/AABII5ePMWdFhAk-dEcRGS1La?dl=0. We use hourly data for January 2016-May 2018 period. We use data from 1 January 2016-26 April 2018 for training and then 27 April-7 May 2018 for testing our model. Figure 10A shows first three months of hourly observations of training load data and shows daily and weekly cycles. Figure 10B shows our hourly training load time series data from 1 January 2016 to 17 April 2018. We can see that demand during summer months is higher compared to winter months. Figure 10A shows the shorter period (January-February 2016) of the same data. We can see that weekends have lower load levels compared to work days.

We develop a feed forward neural network to predict the load for the 5-hour horizon and use previous 24 observations (one day) as predictors. We develop two models, the first is DL model with Gaussian loss function (MSE Loss) and the second is DL-EVT with generalized Pareto loss function (GPar loss). Figure 11 shows the architecture used to model the relations between previous load observations (x) and the scale parameter of the Generalize Pareto distribution σ and ξ. We use recent 24 hours of load values as our predictor vector x.

where $W_1 \in R^{p\times 3}$ and $W_2 \in R^{2\times 3}$, and $x \in R^p$ is the vector of recent observations of electricity demand, we used $p = 24$ (1 day). We use tanh to constrain values of ξ to be in the $(-1,1)$ interval. One of the properties of generalized Pareto random variable is that it has infinite kth moment when $\xi \geq 1/k$. Thus, it has infinite mean when $\xi \geq 1$. Models with infinite mean are applicable for risk analysis[52] and it can be shown[53] that estimated value of ξ can be greater than 1, even when underlying data-generating distribution has finite mean. However, we use expected value as our point forecast of load peaks, and thus, we require $\xi < 1$ to guarantee that we have a finite mean. Furthermore, we require $\xi > -1$ to guarantee that the likelihood function is bounded.

To train the EVT model, we only used the observations $y_i > u$ with $u = 31\,000$. We used the mean $\sigma/(1-\xi) + u$ of the generalized Pareto distribution as the point estimate for plotting Figure 12B.

Our DL-EVT model is compared with a vanilla DL model with standard MSE loss function. Figure 12 shows the resulting out-of-sample forecasts. We can see that while a standard DL model captures both ups and downs in the load

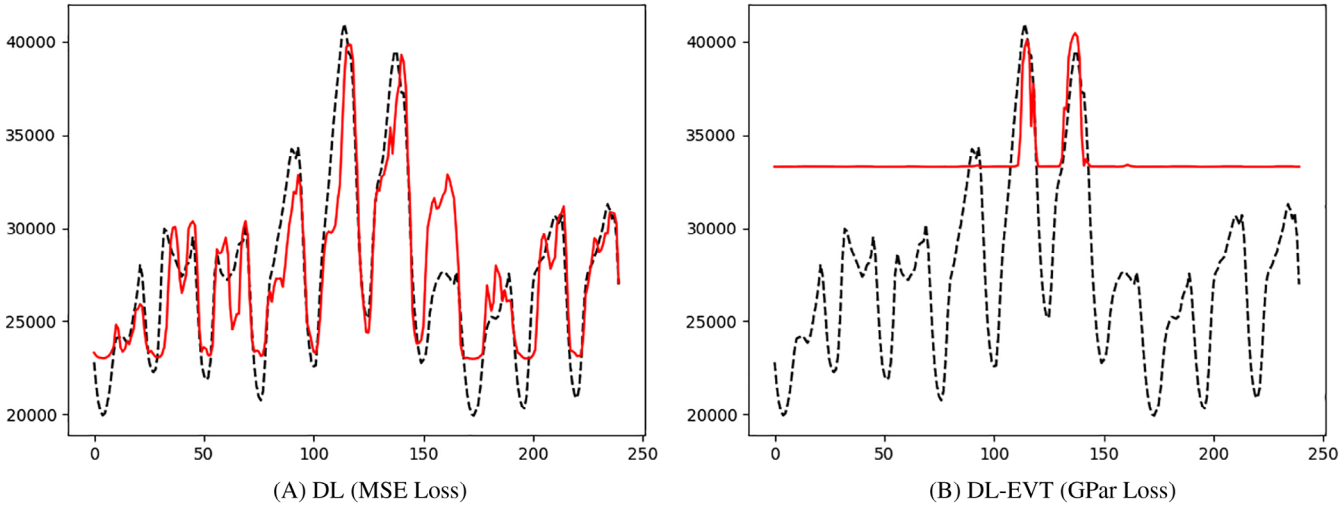


FIGURE 12 Hourly electricity load and its forecast for the period from Friday, 27 April 2018 to Monday, 7 May 2018. The *x*-axis is the hour since 27 April 2018 and the *y*-axis is the load measured in MW. A, DL (MSE Loss). B, DL-EVT (GPar Loss) [Color figure can be viewed at wileyonlinelibrary.com]

levels, the DL-EVT model does capture the location and level of the peak loads more accurately compared to the standard DL model.

6 | DISCUSSION

DL, combined with EVT, can predict peaks in electricity prices and demand. With the availability of RT data, computational power, and machine learning pattern recognition tools, such as DL, we have the ability to more accurately predict and manage energy generation and distribution. One of our goals is to demonstrate that an EVT extension of the standard DL framework is a viable option and is applicable to electricity data. DL-EVT performed well on in- and out-of-sample forecasting of electricity prices and load. We demonstrated empirical performance of our models by predicting price and load for a single node of the PJM system.

We demonstrated our DL-EVT model is more accurate at forecasting peak values that exceed a given threshold when compared to a Gaussian likelihood-based model. The EVT model predicts peak values conditional on the exceedance over the threshold. One of the artifacts of our model is that prediction for time points when a threshold is not expected to be exceeded is a constant value. An extension of our approach could include a binary classifier that predicts the probability of crossing a specific threshold. Another extension is to include a Gaussian likelihood-based model to forecast values below a threshold. Naveau et al[54] demonstrates that a similar approach can be used for successful environmental modeling.

Forecasting electricity prices are challenging because they can spike due to supply-demand imbalances, yet have long-range dependence. Deep ReLU LSTM models capture spikes with nonlinear activation functions, are scalable, and can efficiently fit using SGD. For a grid of 4786 electricity load nodes, we show how such models can fit in-sample with better accuracy than traditional time series models. There are a number for directions of future research. For extensions to multivariate time series data with spatiotemporal dynamics, see Dixon et al.[3]

ORCID

Vadim Sokolov https://orcid.org/0000-0002-6618-2965

REFERENCES

1. Varaiya PP, Wu FF, Bialek JW. Smart operation of smart grid: risk-limiting dispatch. *Proc IEEE*. 2011;99(1):40-57.
2. Hogg RV, Klugman SA. On the estimation of long tailed skewed distributions with actuarial applications. *J Econ*. 1983;23(1):91-102.
3. Dixon MF, Polson NG, Sokolov VO. Deep learning for spatio-temporal modeling: dynamic traffic flows and high frequency trading. *Appl Stoch Model Bus Ind*. 2019;35:788-807.

4. Polson NG, Sokolov VO. Deep learning for short-term traffic flow prediction. *Transp Res Part C Emerg Technol*. 2017;79:1-17. https://doi.org/10.1016/j.trc.2017.02.024.
5. Davison AC, Padoan SA, Ribatet M. Statistical modeling of spatial extremes. *Stat Sci*. 2012;27(2):161-186.
6. Davison AC, Smith RL. Models for exceedances over high thresholds. *J Royal Stat Soc Ser B (Methodol)*. 1990;5(3):393-442.
7. Smith RL. Measuring risk with extreme value theory. *Risk Management: Value at Risk and Beyond*. Vol 224. Cambridge, MA: Cambridge University Press; 2002.
8. Sigauke C, Verster A, Chikobvu D. Extreme daily increases in peak electricity demand: tail-quantile estimation. *Energy Policy*. 2013;53:90-96.
9. Shenoy S, Gorinevsky D. Risk adjusted forecasting of electric power load. Paper presented at: Proceedings of the 2014 American Control Conference; 2014:914-919; IEEE.
10. Catalão JPDS, Mariano SJPS, Mendes VMF, Ferreira LAFM. Short-term electricity prices forecasting in a competitive market: a neural network approach. *Electr Power Syst Res*. 2007;77(10):1297-1304.
11. Hong YY, Hsiao CY. Locational marginal price forecasting in deregulated electricity markets using artificial intelligence. *IEE Proc Gener Transm Distrib*. 2002;149(5):621-626.
12. Hong Ying Yi, Hsiao Chuan-Yo. Locational marginal price forecasting in deregulated electric markets using a recurrent neural network. Paper presented at: Proceedings of the 2001 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No. 01CH37194); 2001:539-544; IEEE.
13. Kim M-K. A new approach to short-term price forecasting strategy with an artificial neural network approach: application to the Nord Pool. *J Electr Eng Technol*. 2015;10:709-718.
14. Mandal P, Senjyu T, Urasaki N, Funabashi T, Srivastava AK. A novel approach to froecast electricity price for PJM using neural network and similar days method. *IEEE Trans Power Syst*. 2007;22:4.
15. Wang L, Zhang Z, Chen J. Short-term electricity price forecasting with stacked denoising autoencoders. *IEEE Trans Power Syst*. 2017;32(4):2673-2681.
16. Hering AS, Genton MG. Powering up with space-time wind forecasting. *J Am Stat Assoc*. 2010;105(489):92-104.
17. Liebl D. Modeling and forecasting electricity spot prices: a functional data perspective. *Ann Appl Stat*. 2013;7(3):1562-1592.
18. Cottet R, Smith M. Bayesian modeling and forecasting of intraday electricity load. *J Am Stat Assoc*. 2003;98(464):839-849.
19. Wilson AL, Dent CJ, Goldstein M. Quantifying uncertainty in wholesale electricity price projections using Bayesian emulation of a generation investment model. *Sustain Energy Grids Netw*. 2018;13:42-55.
20. Jónsson T, Pinson P, Nielsen HA, Madsen H, Nielsen TS. Forecasting electricity spot prices accounting for wind power predictions. *IEEE Trans Sustain Energy*. 2013;4(1):210-218.
21. Christensen TM, Hurn AS, Lindsay KA. Forecasting spikes in electricity prices. *Int J Forecast*. 2012;28(2):400-411.
22. Benth FE, Schmeck MD. Pricing futures and options in electricity markets. *The Interrelationship Between Financial and Energy Markets*. Berlin, Germany: Springer; 2014:233-260.
23. Dupuis DJ. Electricity price dependence in New York state zones: a robust detrended correlation approach. *Ann Appl Stat*. 2017;11(1):248-273.
24. Garcia RC, Contreras J, Akkeren M, Garcia JBC. A GARCH forecasting model to predict day-ahead electricity prices. *IEEE Trans Power Syst*. 2005;20(2):867-874.
25. Li G, Liu C-C, Mattson C, Lawarrée J. Day-ahead electricity price forecasting in a grid environment. *IEEE Trans Power Syst*. 2007;22(1):266-274.
26. Subbayya S, Jetcheva JG, Chen WP. Model selection criteria for short-term microgrid-scale electricity load forecasts. Paper presented at: 2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT); 2013:1-6; IEEE.
27. Kolmogorov AN. On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Dokl Akad Nauk*. 1957;114(5):953-956. [English translation: *Am Math Soc Trans*. 1963; 28(2):55–59].
28. Kolmogorov AN. On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables. *Dokl Akad Nauk SSSR*. 1956;108:179-182. [English translation: *Am Math Soc Trans*. 1961; 17:369–373].
29. Arnold Vladimir I. On functions of three variables. *Dokl Akad Nauk SSSR*. 1963, 14;(4):679-681. [English translation: Am Math Soc Trans. 1963; 14:51–54].
30. Hochreiter S, Schmidhuber J. LSTM can solve hard long time lag problems. *Adv Neural Inf Process Syst*. 1997;473-479.
31. Schmidhuber J, Hochreiter S. Long short-term memory. *Neural Comput*. 1997b;9(8):1735-1780.
32. Bach F. Breaking the curse of dimensionality with convex neural networks. *J Mach Learn Res*. 2017;18(19):1-53.
33. Schmidt-Hieber J. Nonparametric regression using deep neural networks with ReLU activation function; 2017. arXiv:1708.06633.
34. Yarotsky D. Error bounds for approximations with deep ReLU networks. *Neural Netw*. 2017;94:103-114.
35. Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. *Science*. 2006;313(5786):504-507.
36. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res*. 2014;15:1929-1958.
37. Janocha K, Czarnecki WM. On loss functions for deep neural networks in classification; 2017. arXiv:1702.05659.
38. Abadi M, Barham P, Chen J, et al. TensorFlow: a system for large-scale machine learning. Paper presented at: Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation; 2016:265-283.
39. Cardoso PEA. *Deep Learning Applied to PMU Data in Power Systems*. Portugal: Unoversidade Do Porto; 2017.
40. Gnedenko B. Sur la distribution limite du terme maximum d'une serie aleatoire. *Ann Math*. 1943;44(3):423-453.

41. Smith RL. Extreme value analysis of environmental time series: an application to trend detection in ground-level ozone. *Stat Sci*. 1989;4(4):367-377.
42. Beirlant J, Goegebeur Y, Segers J, Teugels JL. *Statistics of Extremes: Theory and Applications*. Hoboken, NJ: John Wiley & Sons; 2006.
43. Polson NG, Sokolov V. Deep learning: a Bayesian perspective. *Bayesian Anal*. 2017;12(4):1275-1304.
44. Polson N, Sokolov V. Bayesian analysis of traffic flow on interstate I-55: the LWR model. *Ann Appl Stat*. 2015;9(4):1864-1888.
45. Cain C, Lesser J, White Bates. *A Common Sense Guide to Wholesale Electric Markets*. Tech Rep April 2007. Bates White Consulting; 2007. http://www.bateswhite.com/media/publication/55_media.741.pdf.
46. FERC Federal Energy Regulatory Commission. Operator-Initiated Commitments in RTO and ISO Markets. FERC Docket No. D14; 2014.
47. PJM. Daily Real-Time LMP: August 1, 2017 through September 7, 2017.
48. Taylor JW, McSharry PE. Short-term load forecasting methods: an evaluation based on european data. *IEEE Trans Power Syst*. 2007;22(4):2213-2219.
49. Garcia MP, Kirschen DS. Forecasting system imbalance volumes in competitive electricity markets. *IEEE Trans Power Syst*. 2006;21(1):240-248.
50. Almeshaiei E, Soltan H. A methodology for electric power load forecasting. *Alex Eng J*. 2011;50(2):137-144.
51. Bunn DW. Short-term forecasting: a review of procedures in the electricity supply industry. *J Oper Res Soc*. 1982;33(6):533-545.
52. Nešlehová J, Embrechts P, Chavez-Demoulin V. Infinite mean models and the LDA for operational risk. *J Operation Risk*. 2006;1(1):3-25.
53. Makarov M. Extreme value theory and high quantile convergence. *J Operation Risk*. 2006;1(2):51-57.
54. Naveau P, Huser R, Ribereau P, Hannart A. Modeling jointly low, moderate, and heavy rainfall intensities without a threshold selection. *Water Resour Res*. 2016;52(4):2753-2769.

How to cite this article: Polson M, Sokolov V. Deep learning for energy markets. *Appl Stochastic Models Bus Ind*. 2020;36:195–209. https://doi.org/10.1002/asmb.2518