

Research Article

Computational Models for Forecasting Electric Vehicle Energy Demand

Mojeed O. Oyedeji ¹, Mujahed AlDhaifallah ^{1,2}, Hegazy Rezk ³,
and Ahmed Ali A. Mohamed⁴

¹Control and Instrumentation Engineering Department, King Fahd University of Petroleum & Minerals,
31261 Dhahran, Saudi Arabia

²Interdisciplinary Research Center (IRC) for Renewable Energy and Power Systems, King Fahd University of Petroleum & Minerals,
31261 Dhahran, Saudi Arabia

³Department of Electrical Engineering, College of Engineering in Wadi Alddawasir,
Prince Sattam bin Abdulaziz University, Saudi Arabia

⁴Department of Electrical Engineering, Grove School of Engineering, City College of the City University of New York, New York,
NY 10031, USA

Correspondence should be addressed to Mujahed AlDhaifallah; mujahed@kfupm.edu.sa

Received 23 August 2022; Accepted 3 September 2022; Published 3 February 2023

Academic Editor: Alexey Mikhaylov

Copyright © 2023 Mojeed O. Oyedeji et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Electric vehicles (EV) are fast becoming an integral part of our evolving society. There is a growing movement in advanced countries to replace gas-driven vehicles with EVs towards cutting down pollution from emissions. When fully integrated into society, electric vehicles will share from energy available on the grid; therefore, it is important to understand consumption profiles for EVs. In this study, some computation models are developed from predicting day-ahead energy consumption for electric vehicles in the city of Barcelona. Five different machine learning algorithms namely support vector regression (SVR), Gaussian process regression (GPR), artificial neural networks (ANN), decision tree (DT), and ensemble learners were used to train the forecasting models. The hyperparameters for each of the ML algorithms were tuned by Bayesian optimization algorithm. In order to propose efficient features for modeling EV demand, two different model structures were investigated, named Type-I and Type-II model. In the instance of the Type-I model, seven regressors representing the consumption of the previous seven days were considered as input features. The Type-II models considered only the EV consumption on the previous day and on the same day in the previous week. Based on the results in this study, we find that the performance of the Type-II models was as good as the Type-I models across all the algorithms considered although less input features were considered. Overall, the all algorithms employed in this study gave about 75-80% model accuracy based on the R^2 performance criterion. The models formulated in this study may prove useful for planning and unit commitment functions in city energy management functions.

1. Introduction

Climate change concerns have stimulated diverse interests in alternative and sustainable sources of energy. The past decade has witnessed significant contributions in research and developments geared towards reduction in the dependence on fossil energy sources and integration of more renewable energy sources into the world's energy mix. Con-

sequently, the generation of energy from renewable sources has led to innovation and development in everyday machines and devices that consume renewable energy. Since it is conceptualization in the early nineteenth century, electric vehicles (EVs) have grown in prominence and adoption. It is estimated that by 2035, all newly purchased vehicles in the EU will be solely electric, due to the ban imposed on the sale of new petrol and diesel cars the EU parliament

[1]. The volume of new EV registration in Europe increased from 3.5% to 11% between 2019 and 2020, while the purchase of electric vans also increased from 1.4% to 2.2% within the same period [2].

In a not-too-distant future, EVs will compete for energy on the grid. The faster the drive for integration of EVs into our society, the higher the energy consumption that will be demanded from grid operators on an hourly-daily time resolution. However, the nature of EV demand can be quite stochastic and therefore present some problems in forecasting. This has driven research interests into developing forecasting models that are capable of predicting consumption. Conversely, due to the current low level of adoption, there is limited available open data to assist the development of efficient EV demand forecasting models.

The development of accurate models for EV demand could offer significant economic benefits by minimizing unit commitment costs through providing reliable forecasts [3–7]. In [8], the authors studied day-ahead charging demand of electric vehicles using a deep-neural network. In their study, they show that the performance measures of the neural network forecaster, namely RMSE and MAE, improved by 28.8% and 19.22%, respectively, due to the inclusion of weather and calendar features in the forecasting problem formulation. An autoregressive integrated moving average (ARIMA) model was proposed in [9] for forecasting charging demand and conventional electrical load of EV charging stations. The proposed forecaster considers daily driving patterns and distances as features to generate the expected charging load profiles. According to the authors, the performance of the ARIMA model was improved by tuning the order of the integrated and autoregressive parameters such that the mean squared error performance criterion is minimized. A Bayesian inference method with convolution was proposed by [10] for 24 hours ahead EV charging demand. The performance of the proposed method was compared with linear regression and was found to have lower MAPE and RMSE values. Different classical and machine learning-based algorithms were considered in [11] for forecasting EV demand in Korea using datasets from 2018 to 2019. In their study, they considered classical statistical methods such as the trigonometric, Box-Cox, autoregressive-moving-average (ARMA), and autoregressive integrated moving average (ARIMA), as well as machine learning methods including artificial neural networks and long short-term memory (LSTM) networks. The study specifically evaluated the influence of exogenous variables in macro- and microscale geographical areas.

A decision tree-based model was proposed in [12] for estimating charge demand for different classes of EVs in South Korea. The proposed model incorporates traffic distribution data and weather effects as features in the forecasting model. The study built different cluster models, using probability density functions, for electric cars and buses based on traffic patterns. Four different prediction methods, namely, Time Weighted Dot Product Nearest Neighbor, Modified Pattern Sequence Forecasting, support vector regression, and random forests were applied to charging record and station record datasets from UCLA in [13]. It was the objective

of the authors to determine the best approach between charging record (consumer-perspective) or station record (charging outlets/stations) to forecast EV charging load. The authors conclude that both approaches yielded comparative prediction errors. Support vector regression was utilized by [14] to build an EV forecaster. The features considered in the study included the following: historical charging data, number of EVs, weather information, week properties, and holiday properties. Short-term EV demand forecasting model was investigated by [15] using a convolutional neural network (CNN) optimized by a niche immunity lion algorithm. The niche immunity lion algorithm was used in their study to optimize the weights and thresholds of the CNN.

Contrary to the aforementioned studies, this study was interested in the development of multiple computational models for EV demand forecasting towards suggesting the best model with the least prediction errors on limited historical data. Furthermore, it was of paramount interest to test whether increasing the number of regressors (features) in time series forecasting for EV demand yielded better model prediction capabilities. The main contributions of this study are summarized as follows:

- (i) We develop computational models for predicting day-ahead EV consumption in the Municipal Area of Barcelona (*Area Metropolitana de Barcelona, AMB*), Spain
- (ii) Five machine learning (ML) algorithms were investigated in this study, namely artificial neural networks (ANN), support vector regression (SVR), Gaussian process regression (GPR), decision tree (DT), and ensemble learners (EL).
- (iii) We investigate two main model structures each differing in number of features considered. To forecast the demand $L(k)$ on a given day k , the Type-I model considered total EV energy demand in the past seven days, that is, $[L(k-1), \dots, L(k-7)]$, while the Type-II model considered only two features, total EV energy in the previous day and on the same day in the previous week
- (iv) The parameters of each of the ML algorithms were optimized using Bayesian optimization to arrive at the optimal set of hyperparameters that give optimal model results
- (v) Performance analyses were carried out for each model cluster using the R^2 regression coefficient and root mean squared error (RMSE) values
- (vi) It was determined that the Type-II models performed just as well as the Type-I models with 75–80% model prediction accuracy across all five ML algorithms considered which suggests that the previous day consumption $L(k-1)$ and consumption on the same day in the previous week $L(k-7)$ represent important features in modeling consumption profiles of EV users in Barcelona

TABLE 1: Average daily energy consumption in each station.

SN	Station	Av. cons. (kWh/d)
1	FLNR Sant Andreu da la Barca: Pg. Rafael de Casanova FGC	11.3427
2	PdRL Badalona: C. Anna Tugas - Pg. Olof Palmer	2.469
3	PdRL BarberÀ del VallÀ©s: C. ArquÀ-medes, 8	0.5813
4	PdRL CornellÀ de Llobregat: Carrer de Baltasar Oriol i Mercer	1.9629
5	PdRL El Prat de Llobregat: Pl. Volateria (Mas Blau)	0.4934
6	PdRL GavÀ : C. del Progres, 54	8.3547
7	PdRL L'Hospitalet de LL: C. Salvador Espriu - Gran Via de les Corts Catalanes	2.3785
8	PdRL Montcada i Reixac: C. Tarragona - C. Pla de Matabous	1.5036
9	PdRL PallejÀ : Rda. Santa Eulalia - C. Joan Maragall	3.4857
10	PdRL Sant Cugat del VallÀ¨s: Av. Via Augusta, 3	2.9999
11	PdRL Sant Joan DespÀ-: C. TV3 - C. Jacint Verdaguer	5.1317
12	PdRR Badalona: C. Anna Tugas - Pg. Olof Palmer	54.5
13	PdRR BarberÀ del VallÀ©s: C. ArquÀ-medes, 8	40.6751
14	PdRR CornellÀ de Llobregat: Carrer de Baltasar Oriol i Mercer	133.5475
15	PdRR El Prat de Llobregat: Pl. Volateria (Mas Blau)	110.5672
16	PdRR GavÀ : C. del Progres, 54	147.3415
17	PdRR Montcada i Reixac: C. Tarragona - C. Pla de Matabous	36.4623
18	PdRR PallejÀ : Rda. Santa Eulalia - C. Joan Maragall	78.5579
19	PdRR Sant Cugat del VallÀ¨s: Av. Via Augusta, 3	150.5473
20	PdRR Sant Joan Despi: C. TV3 - C. Jacint Verdaguer	109.0142
21	PdRR L'Hospitalet de LL: C. Salvador Espriu - Gran Via de les Corts Catalanes	119.3131

2. Datasets

The dataset considered in this study was obtained from [16] and represents the charge demand from EV users in the city of Barcelona covering a full one year period in 2019. The dataset consists of records from 21 stations. Each of the stations and their average daily energy consumption are summarized in Table 1. The first 11 stations in Table 1 represent slow charging stations with two Schuko 3kW charging points (CPs). The last 10 stations represent fast charging stations each with the following charging points: one Mennekes 43kW CP, one Combo CCS 55kW CP, and one CHAdeMO 55kW CP. Figure 1 summarizes the average demand for each day type, that is, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday which shows that bulk of the EV energy consumption occurs during weekdays than weekends.

3. Methodology

Two model structures were formulated to investigate the consumption patterns of EV users in the city of Barcelona. Type-I model uses seven predictors representing the consumption of the past week, that is, $[L(k-1), \dots, L(k-7)]$, while Type-II model considers just two features, $[L(k-1), L(k-7)]$ denoting the consumption of the previous day and the consumption on the same day last week. Both model types were trained with five machine learning algorithms:

Decision tree, ensemble learners, support vector regression, Gaussian process regression, and artificial neural networks. For each of the algorithms, we carry out hyperparameter optimization using Bayesian optimization to select the best hyperparameters. The features of both model types were extracted in the same manner from the original data and split into training and testing set using a 70:30 split ratio. The following subsections discuss each of the machine learning algorithms considered in the study.

3.1. Decision Tree. Decision trees are graph-based data mining algorithms for regression and classification problems. The earlier decision tree algorithms such as the Iterative Dichotomiser III [17] and C4.5 [18, 19] were proposed for classification-based problems and introduction of CART [20] which is capable of solving regression problems. In a decision tree algorithm, each interior node represents an input variable, and each terminal node corresponds to a target variable. The regression (decision) trees are grown to minimize errors by dividing the input space at each iteration of the algorithm. The estimated output at each terminal node is computed by

$$\hat{y}_i = \frac{1}{t_i} \sum_{j \in t_i} y_j, \quad (1)$$

where t_i represents the leaf node at i and $|t_i|$ is the total number of samples at that node. An error criterion is used to

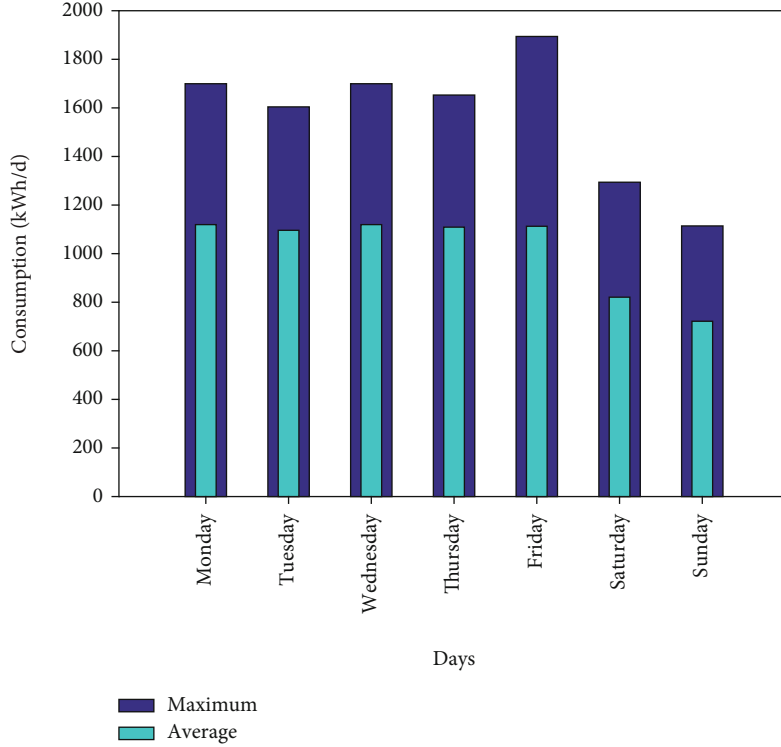


FIGURE 1: Maximum and average consumption by day type.

guide the splitting decision. For instance, [] considers the following least squares deviation

$$I(t_i) = \sum_{j \in t_i} (y_j - \hat{y}_i)^2, \quad (2)$$

where $I(t_i)$ denotes an impurity measure. The splitting criterion is then obtained as follows.

$$\Delta I = I(t_p) - P_l I(t_l) - P_r I(t_r). \quad (3)$$

t_l and t_r represent child nodes to the left and right of the parent node t_p , respectively, and P_l and P_r are data proportions of samples allocated to left and right children nodes. Suppose the split rule is generated based on numeric or ordinal variable, the resulting children node is two, and the parent node is decomposed into two subsets, $\{x : x_k > S\}$ and $\{x : x_k \leq S\}$, where s , x_k denotes the split point and selected attribute, respectively. Important hyperparameters considered in this study which could affect the performance of the decision tree algorithm include maximum number of splits, minimum leaf size, and number of variables to sample. The maximum number of splits parameter limits the total number of splits possible from the root node. The number of variables to sample parameter determines the number of regressors selected at each random split. The minimum leaf size hyperparameter defines the limit for a node split in a child node when the number of observations in that node is less than the minimum leaf size.

TABLE 2: Model performance measures.

Measure	SVR	GPR	ANN	DT	Ensemble
Type-I					
R^2 (training)	0.807	1	0.809	0.832	0.89
R^2 (testing)	0.79	0.79	0.796	0.664	0.806
RMSE (training)	172.44	0.05	171.366	161.963	135.41
RMSE (testing)	189.01	189.78	187.719	234.5571	183.89
Type-II					
R^2 (training)	0.802	1	0.8048	0.81	0.83
R^2 (testing)	0.803	0.787	0.802	0.75	0.77
RMSE (training)	174.11	0.05	173.67	170.15	160.87
RMSE (testing)	184.42	190.89	185.12	202.87	198.53

3.2. Ensemble Learners. Voting ensemble or ensemble learning refers to the combination of multiple algorithms to solve a given problem. This approach has been proven to optimize performance, reduce instability, and handle complex datasets. Different ensemble-based techniques have been reported in literature such as voting-based ensemble [21], ensemble of online sequential extreme learning machine (EOS-ELM) [22], and weighted voting in ELM [23]. Voting-based ELM considers multiple ELMs trained on the same dataset, each having the same hidden node and activation function in each hidden node. A majority-based voting is then used to determine the final output of the ensemble network. In this study, we consider ensemble learning

TABLE 3: SVR Hyperparameter optimization results.

Model	BC	KS	Epsilon	KF	PO	Standardize	MO
SVR-I	9.3477	—	1.2361	<i>Linear</i>	—	<i>True</i>	10.32
SVR-II	131.22	3.2215	45.721	<i>Gaussian</i>	—	<i>True</i>	10.3678

BC = box constraint; KF = kernel function. KS = kernel scale; PO = polynomial order; MO = minimum objective.

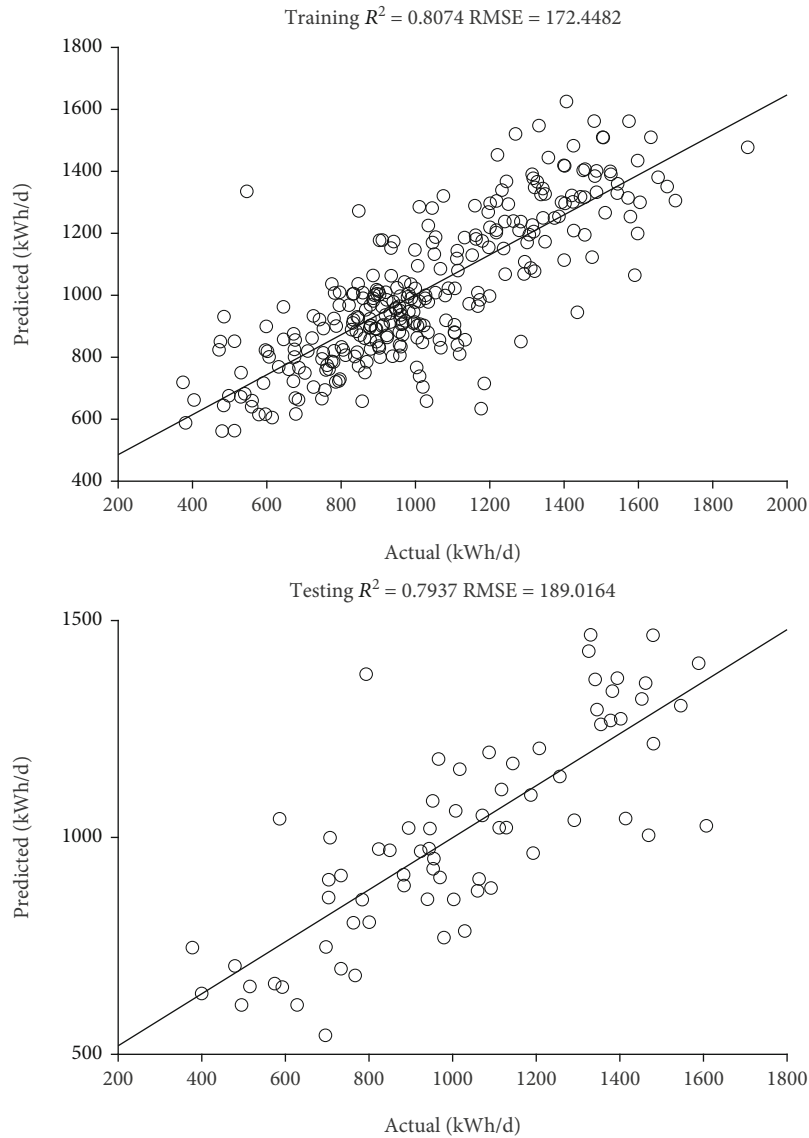


FIGURE 2: Training and testing regression plots for SVR-I model.

algorithm consisting of boosted or bagged decision trees. The optimized hyperparameters of the ensemble learning method considered in this study include the following: method (LSBoost or Bag), number of learning cycles, learn rate, minimum leaf size, and number of variables to sample. The minimum leaf size, number of learning cycles, and number of variables to sample are particular to the decision tree algorithm and have been discussed in the preceding section. The method hyperparameter specifies the type of ensemble method to use. Boosting methods constructs shal-

low decision trees and are faster to converge than Bag method which constructs deep trees. The learn rate hyperparameter guides the speed of convergence of the LSBoost algorithm.

3.3. *Support Vector Regression.* Support vector machine (SVM) originally proposed by [24] is based on the structural risk minimization principle. Given a dataset in a binary classification task, the classical SVM algorithm attempts to find a hyperplane that minimizes the

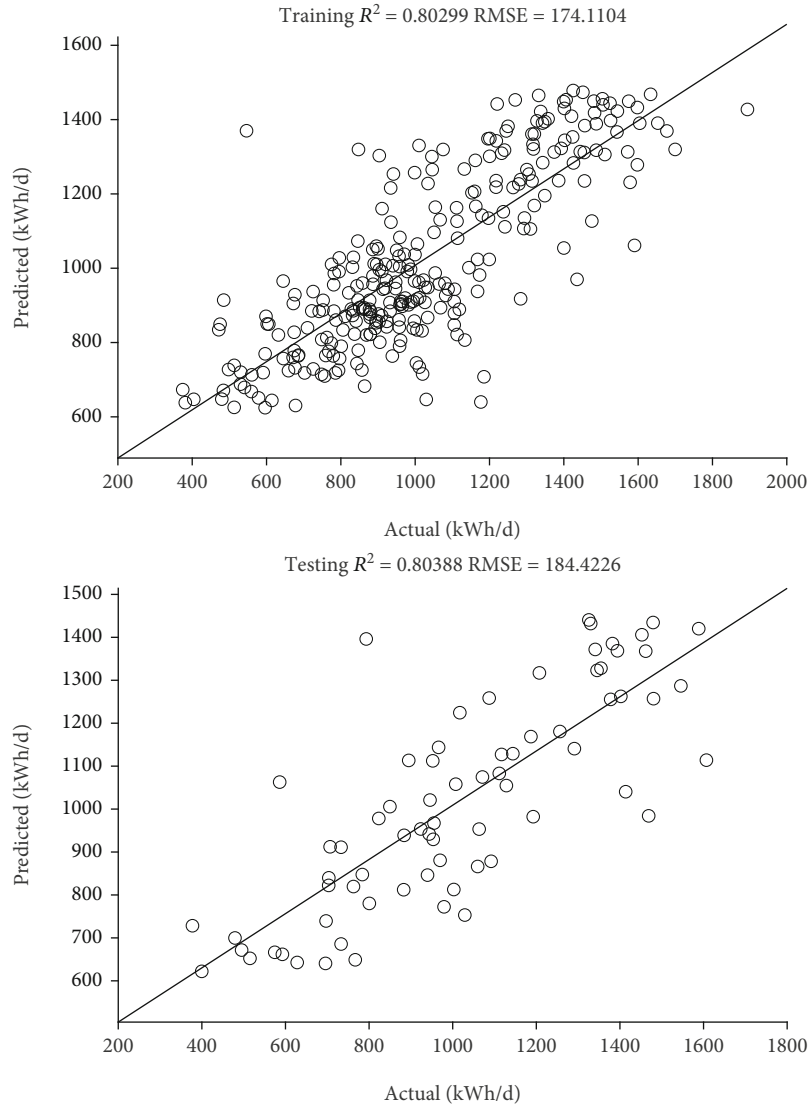


FIGURE 3: Training and testing regression plots for SVR-II model.

TABLE 4: GPR hyperparameter optimization results.

Model	Sigma	BF	KF	KS	Standardize	MO
GPR-I	0.013969	<i>Linear</i>	<i>Ardexponential</i>	—	<i>True</i>	10.355
GPR-II	1.7×10^{-4}	<i>Linear</i>	<i>Exponential</i>	1.608	<i>False</i>	10.355

BF = basis function; KF = kernel function. KS = kernel scale; MO = minimum objective.

classification error. Support vector regression (SVR) algorithms are extensions of the SVM algorithm for regression-based problems. The classical SVR algorithm is based on a ϵ -loss function which attempts to compute a hyperplane such that the error between the predicted observations \hat{y} , and the actual observations y based on a given feature space is minimized to have at least ϵ -deviation. Different variants of the SVR algorithms have been proposed such as linear, kernel, V, and twin-SVR algorithms. A simple mathematical basis of the SVR algorithm

is summarized as follows. Consider a simple regression problem with n observations arranged in the tuple (x_i, y_i) , where x_i and y_i denote features (or predictors or regressors) and outputs, respectively. Suppose we seek to find a function $f(x)$ defined by

$$f(x) = \omega x + b, \quad (4)$$

such that the error between the observed outputs y and the predicted outputs $f(x)$ is minimized over training

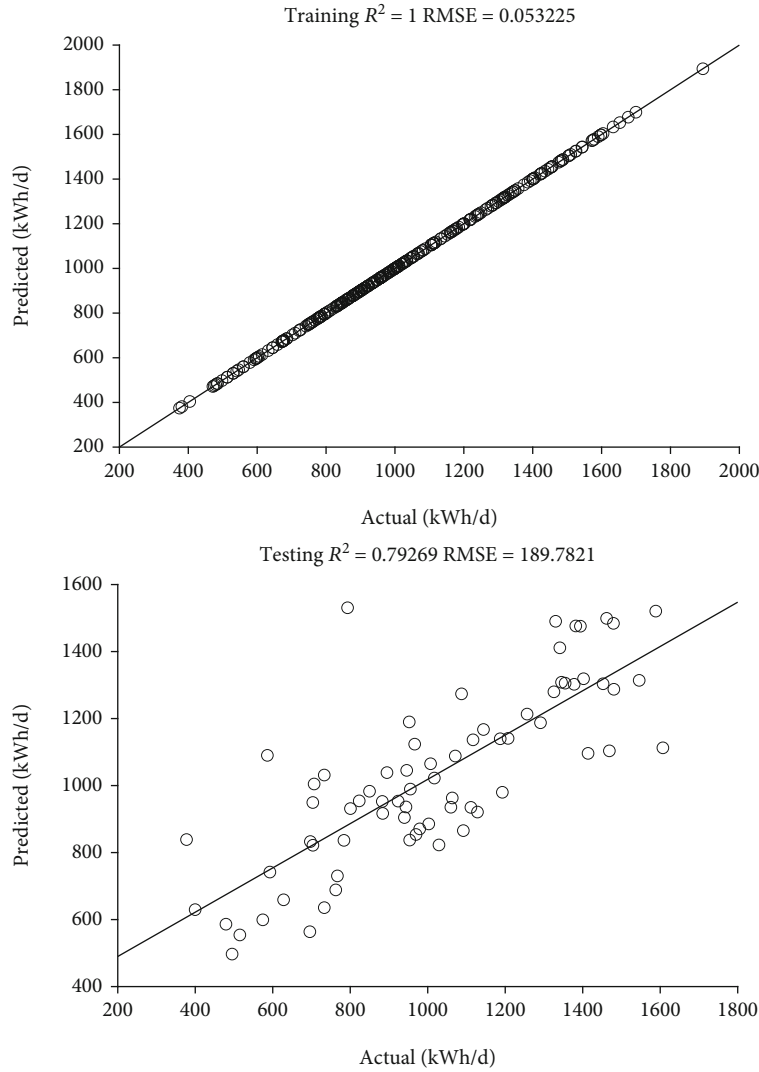


FIGURE 4: Training and testing regression plots for GPR-I model.

and validation datasets. The problem is thus cast into an optimization problem as follows.

$$\begin{aligned} & \min_{\omega} \frac{1}{2} |\omega|^2 \\ & \text{subject to} \begin{cases} y_i - \omega x_i - b \leq \varepsilon \\ \omega x_i + b - y_i \leq \varepsilon. \end{cases} \end{aligned} \quad (5)$$

Alternatively, we may also consider the following loss functions [Reference]

$$L(y, f(x)) = \begin{cases} 0 & |y - f(x)| \leq \varepsilon, \\ |y - f(x)| - \varepsilon & \text{otherwise,} \end{cases} \quad (6)$$

$$L(y, f(x)) = \begin{cases} 0 & |y - f(x)| \leq \varepsilon, \\ (|y - f(x)| - \varepsilon)^2 & \text{otherwise.} \end{cases} \quad (7)$$

For practical considerations, the presented mathematical formulation of the SVR algorithm is too naïve as it does not account for model inaccuracies. To account for model errors, slack variables ζ_1, ζ_2 are introduced, to reconstruct the minimization problem as follows:

$$\begin{aligned} & \min_{\omega} \frac{1}{2} |\omega|^2 + C \sum_{i=1}^l (\zeta_{i,1} + \zeta_{i,2}) \\ & \text{subject to} \begin{cases} y_i - \omega x_i - b \leq \varepsilon + \zeta_{i,1} \\ \omega x_i + b - y_i \leq \varepsilon + \zeta_{i,2} \\ \zeta_{i,1}, \zeta_{i,2} \geq 0. \end{cases} \end{aligned} \quad (8)$$

In (8), a positive regularization parameter has been included to balance between optimizing the flatness of the function f and minimizing the prediction errors. The kernel SVR is arguably the most popular SVR algorithm in literature due to its ability to handle highly nonlinear problems by transforming the input space into high-

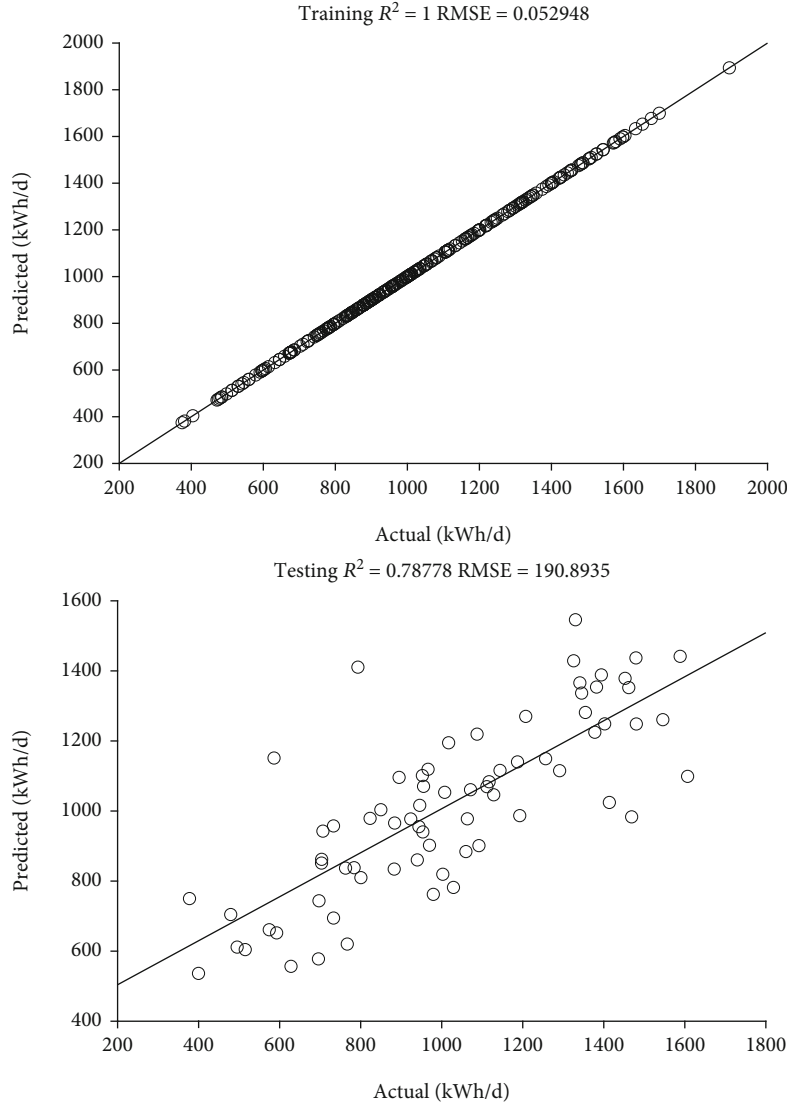


FIGURE 5: Training and testing regression plots for GPR-II model.

TABLE 5: ANN Hyperparameter optimization results.

Model	NL	Activations	Standardize	Lambda	LWI	LBI	L_1	L_2	L_3	MO
ANN-I	1	<i>ReLU</i>	<i>True</i>	0.0791	<i>He</i>	<i>Zeros</i>	297	—	—	10.29
ANN-II	3	<i>ReLU</i>	<i>True</i>	8.5295×10^{-5}	<i>He</i>	<i>Zeros</i>	2	15	2	10.34

NL = number of layers; LWI = layer weight initializer; LBI = layer bias initializer; L_i = size of layer $i = 1, 2, \dots, NL$; MO = minimum objective.

dimensional kernel space via some kernel functions. For a given kernel function $\phi(\cdot) \in \mathbb{R}^m \rightarrow \mathbb{R}$, the kernel SVR problem is cast as follows:

$$\min_{\omega} \frac{1}{2} |\omega|^2 + C \sum_{i=1}^l (\zeta_{i,1} + \zeta_{i,2})$$

$$\text{subject to} \begin{cases} y_i - \omega \phi(x_i) - b \leq \varepsilon + \zeta_{i,1} \\ \omega \phi(x_i) + b - y_i \leq \varepsilon + \zeta_{i,2} \\ \zeta_{i,1}, \zeta_{i,2} \geq 0. \end{cases} \quad (9)$$

The performance of the support vector regression algorithm may be influenced by the choice of hyperparameters. Some tunable SVR hyperparameters include the following: box constraint, kernel scale, epsilon, kernel function, and polynomial order. The box constraint hyperparameter controls the costs associated to misclassified points when the data is not linearly separable. The kernel function hyperparameter determines the type of kernel function to interpret the features. Common kernel functions include linear, Gaussian, radial basis, polynomial, and sigmoidal functions. The margin of tolerance is measured by the epsilon hyperparameter. The

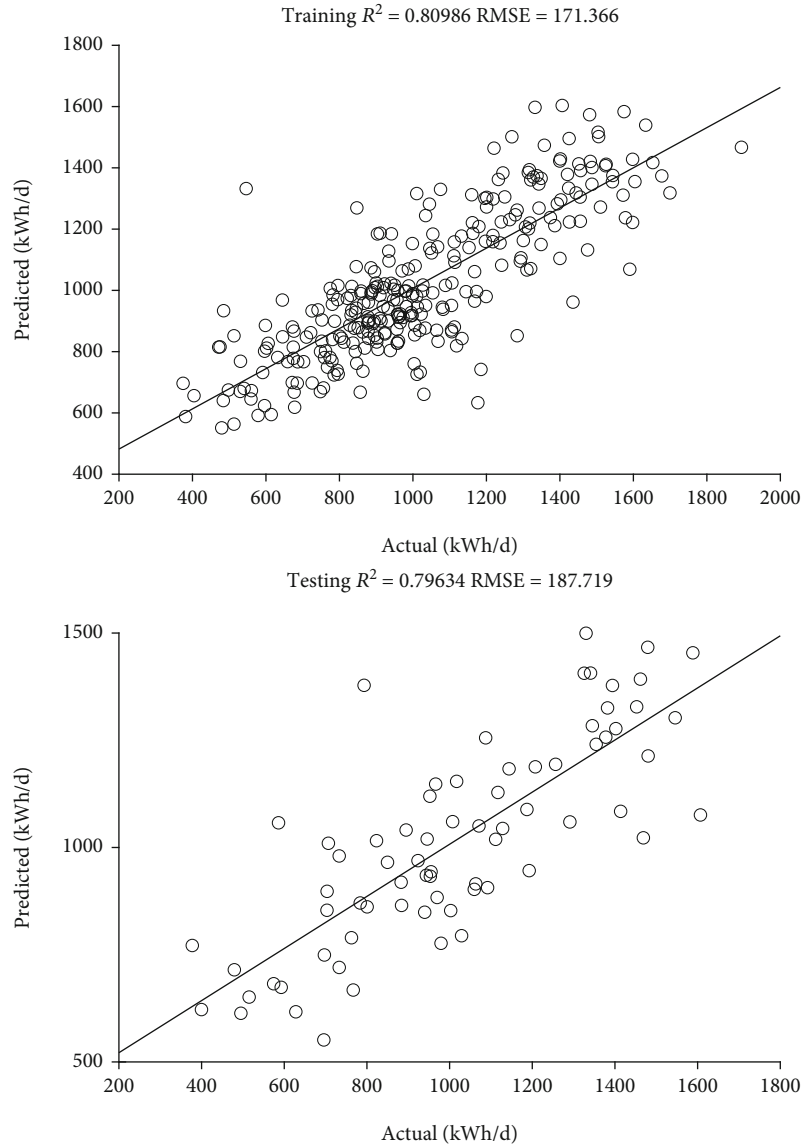


FIGURE 6: Training and testing regression plots for ANN-I model.

smaller the value of ϵ , the smaller the error tolerance. The polynomial order hyperparameter determines the order of the polynomial kernel function. The kernel scale hyperparameter is a scaling value used to scale the features before computation of the Gram matrix.

3.4. Artificial Neural Networks. Artificial neural networks (ANN) are bioinspired information processing units that mimic the processing capabilities of the human brain to model complex and often highly nonlinear relationships between input and output data. ANN was originally developed by [25], in their attempt to derive functional models for biosystems using simple logical operators. The study and development of artificial neural networks has since then evolved with several model structure and algorithms proposed for solving different black-box modeling problems. The simplest neural network consists typically of a neuron

which forms its basic functional units. The mathematical expression of a simple neuron unit is given by

$$\hat{y} = \phi \left(\sum_{i=1}^n \omega_i x_i + b \right), \quad (10)$$

where ω_i and b_i are known as weights and biases, \hat{y} is the output of the neuron, and n is the number of data observations. The process of arriving at black-box models for complex and/or nonlinear processes is referred to as learning. In the learning phase, the neural networks adopt some learning algorithms to determine appropriate weights. A loss function usually in the form of mean squared error criterion between the actual and predicted output is defined to guide the convergence of the learning algorithms to optimal values of the weights and biases that fits the modeling problem

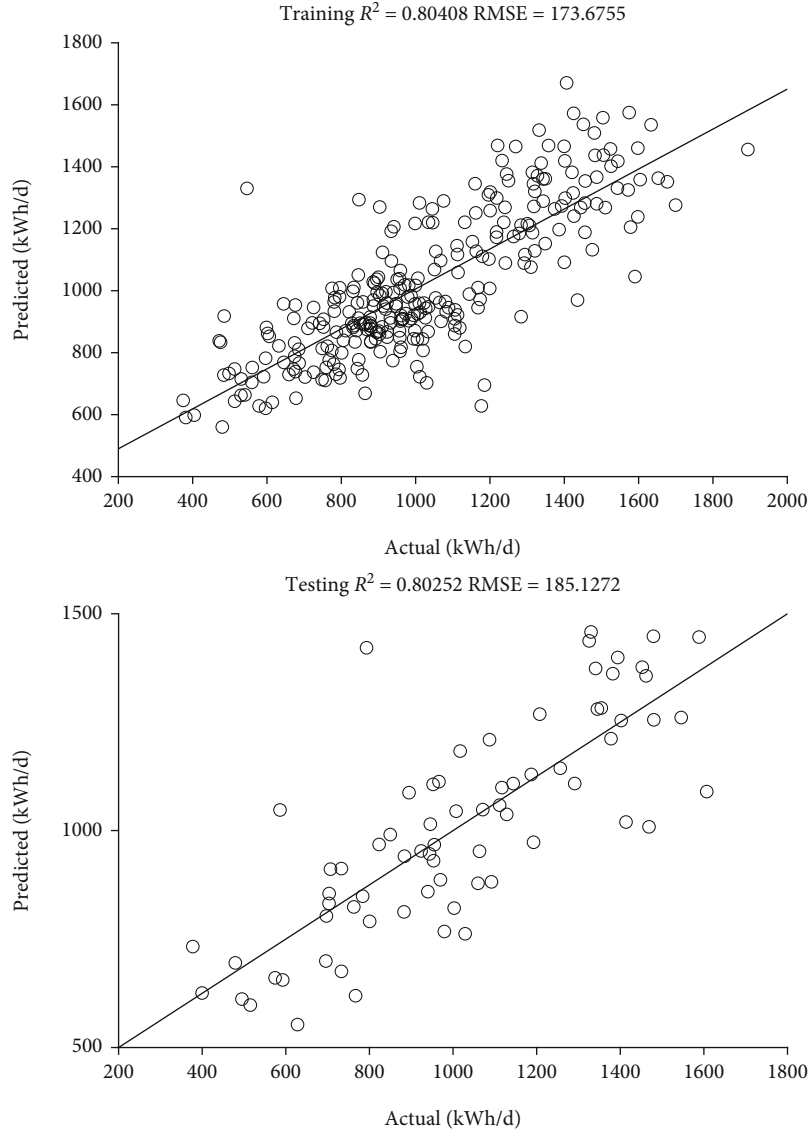


FIGURE 7: Training and testing regression plots for ANN-II model.

perfectly. Different forms of learning such as supervised, unsupervised, and reinforcement learning have been proposed for neural network modeling problems. Learning algorithms such as gradient descent, Levenberg Marquardt, newton, quasi-Newton, and conjugate gradient have been developed for training neural network models. ANNs can be classified based on the network structure into single-layer feed-forward networks, multilayer feed-forward networks, single-layer recurrent network, multilayer recurrent network, and single node with self-feedback. ANNs have the capabilities to handle both classification and regression problems. In this study, we employ a feed-forward neural network model structure with a limited memory Broyden-Fletcher-Goldfarb-Shanno quasi-Newton (LBGGS) training algorithm to minimize the mean squared error criterion. The layer structure of the neural network consists of five layers, namely the input layer, fully connected layer, ReLU activation layer, another fully connected layer, and an output layer. The

TABLE 6: Decision tree hyperparameter optimization results.

Model	MLS	MNS	VS	MO
DT-I	4	7	7	10.41
DT-II	27	5	2	10.40

MLS = minimum leaf size; MNS = maximum number of splits; VS = number of variables to sample; MO = minimum objective.

main hyperparameters optimized for the ANN algorithms include the activation function, the number of hidden layers, and the number of neurons in each hidden layer.

3.5. *Gaussian Process Regression.* The approach for modeling complex and highly nonlinear processes via Gaussian process regression (GPR) is based on statistical Gaussian processes defined by a mean and covariance function. Consider a process to be modeled which has the pair of input-output relationships defined by $(x_i, y_i), i = 1, \dots, n$,

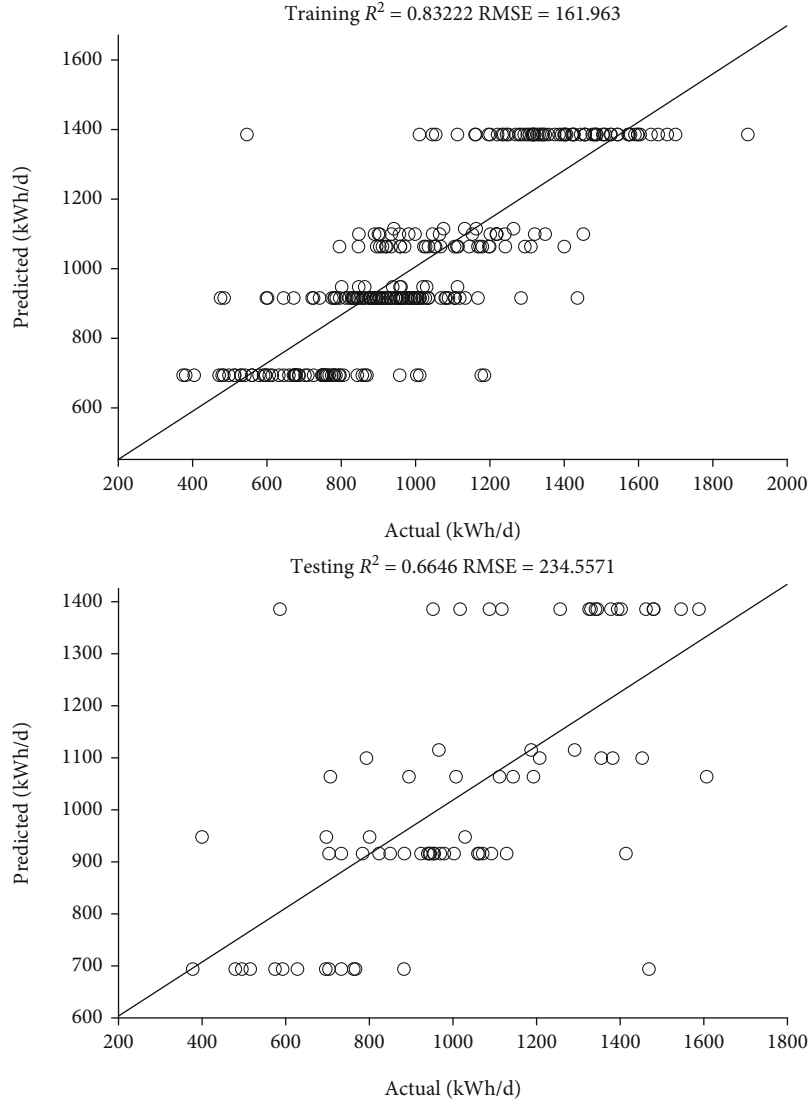


FIGURE 8: Training and testing regression plots for DT-I model.

where $x_i \in \mathbb{R}^m$, $y_i \in \mathbb{R}$ represent features (input) and outputs, respectively, and n is the number of observations. The modeling in GPR begins with describing a predictor \hat{y} for the process output y as follows:

$$\hat{y} = f(x) + \varepsilon, \quad (11)$$

where $\varepsilon \approx \mathcal{N}(0, \sigma_\varepsilon^2)$ is a noise term with zero mean and covariance σ_ε^2 . The function $f(x)$ is also considered to be random variable and can be described by a statistical distribution, in this case, a Gaussian distribution as follows:

$$f(x) \approx GP\left(m(x), k(x, x')\right), \quad (12)$$

where $m(x)$ and $k(x, x')$ are mean and kernel (covariance) functions defined as follows:

$$m(x) = \mathbb{E}(f(x)), \quad (13)$$

$$k(x, x') = \mathbb{E}(f(x) - m(x))(f(x') - m(x')). \quad (14)$$

The kernel functions are used to describe the dependence of the functions at different points x and x' and are critical to the performance of the GPR algorithm in a modeling task. A popular choice of kernel function is the Bayesian kernel function defined as

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{|x - x'|^2}{2\lambda^2}\right), \quad (15)$$

where λ and σ_f^2 are length scale and signal variance hyperparameters, respectively, and are used to increase or decrease prior correlation between points. The major hyperparameters optimized for the GPR algorithm include basis function, sigma, kernel function, and kernel scale. The options of GPR kernel functions include *ardexponential*, *ardmatern32*, *ardmatern52*, *ardrationalquadratic*, *ardsquaredexponential*,

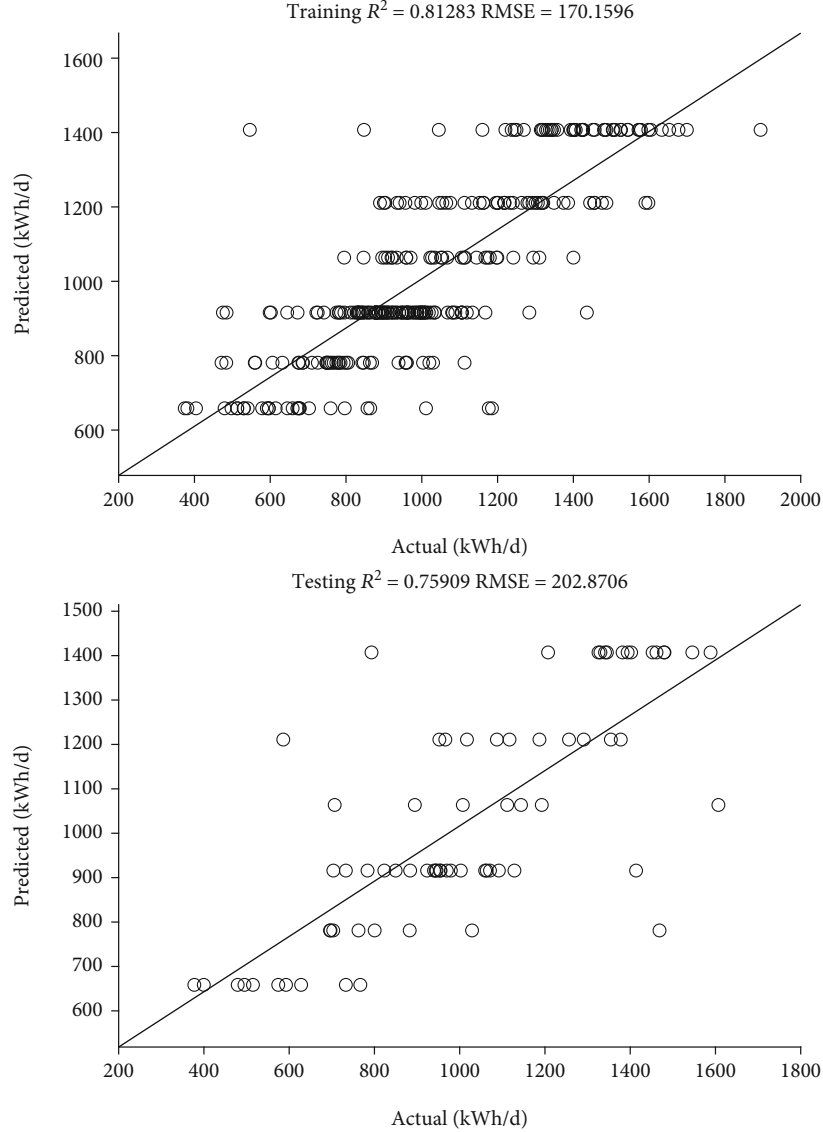


FIGURE 9: Training and testing regression plots for DT-II model.

TABLE 7: Ensemble hyperparameter optimization results.

Model	Method	NLC	LR	MLS	MNS	VS	MO
Ensemble-I	Bag	16	—	19	31	5	10.34
Ensemble-II	LSBoost	39	0.28721	21	1	2	10.36

MLS = minimum leaf size; MNS = maximum number of splits; VS = number of variables to sample; NLC=number of learning cycles. LR= learning rate; MO = minimum objective.

exponential, matern32, matern52, rationalquadratic, and squaredexponential. The search space of the basis function includes *none, constant, linear, and quadratic.*

4. Results and Discussion

This section presents the results obtained from both the Types-I and II demand forecasting models. The performances of each computational model are evaluated using

root mean squared error (RMSE) and correlation coefficient (R^2) defined by (16) and (17), respectively. Table 2 summarizes the performance values for all the models considered.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (16)$$

$$R^2 = \left(\frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}} \right)^2. \quad (17)$$

4.1. Support Vector Regression. The optimized hyperparameters of the SVR model are reported in Table 3. The Type-I SVR model optimization converged to a linear kernel function, while for the Type-II model, the optimizer converged

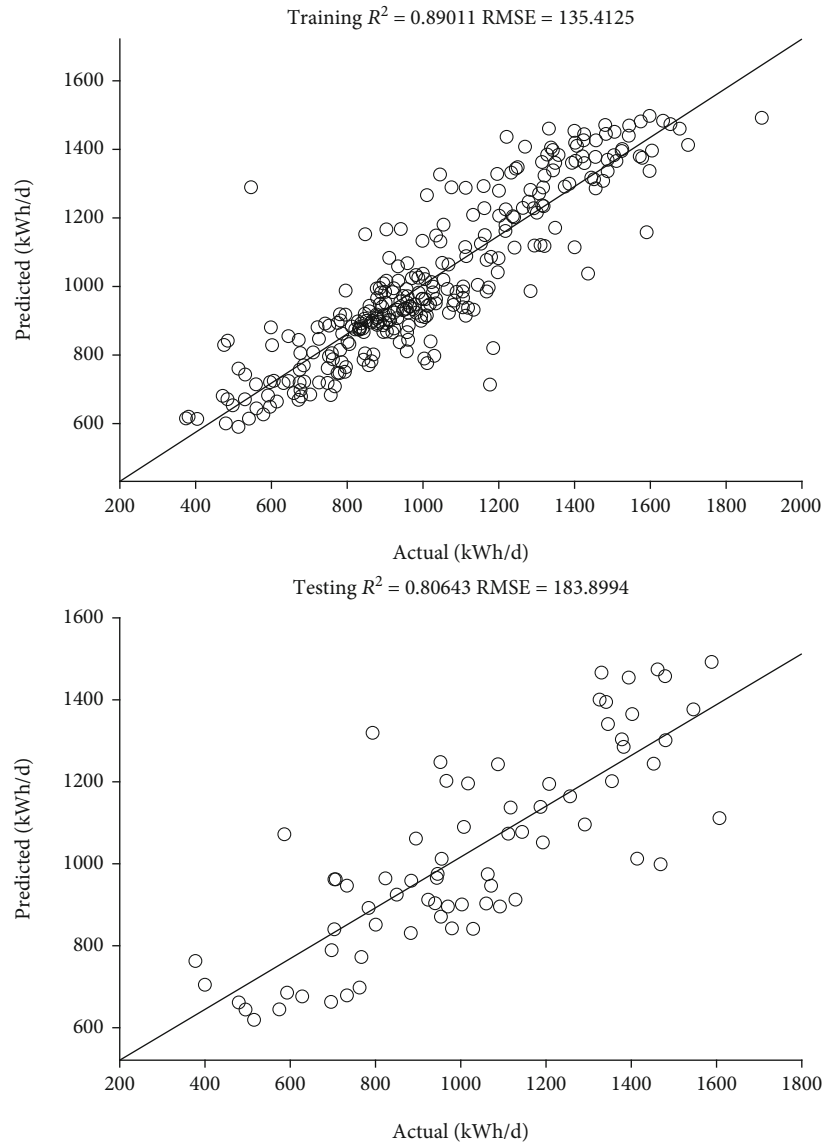


FIGURE 10: Training and testing regression plots for ensemble-I model.

to a Gaussian kernel function. The Type-I SVR model converged to a minimum objective of 10.32, while the Type-II SVR model converged to a minimum objective of 10.36. Figures 2 and 3 show the training and testing regression line plots for both Type-I and Type-II models. The evaluation of the Type-I SVR algorithm on the training dataset yielded R^2 and RMSE values of 0.80 and 172.44, respectively, while the Type-II model yielded R^2 and RMSE values of 0.80 and 174.11 on the training set. Likewise, the Type-I model gave R^2 and RMSE values of 0.79 and 189.01, respectively. On the testing set for the Type-II model, we recorded R^2 and RMSE values of 0.80 and 184.42, respectively.

4.2. Gaussian Process Regression. Table 4 presents the results of the optimized GPR hyperparameters. The Type-I model converged to a linear basis function and arDEXponential kernel function, while the Type-II model converged to a linear basis function and an exponential kernel function. The minimum

objective function values for both Type-I and Type-II models are 10.35 and 10.35, respectively. In Figures 4 and 5, the regression plots of Type-I and Type-II models are presented. In both Type-I and Type-II model formulations, R^2 and RMSE values of 1 and 0.05 were recorded on the training dataset. Conversely, the testing performance for the Type-I model gave R^2 and RMSE values of 0.79 and 189.78, respectively. Likewise, R^2 and RMSE values of 0.78 and 190.89 were recorded on the training set for Type-II model.

4.3. Artificial Neural Networks. In Table 5, we summarize the results of the hyperparameter optimization of the ANN algorithm. In the case of Type-I model, the optimization algorithm converged to one hidden layer with 297 neurons. Conversely, in the case of Type-II model the optimization algorithm converged to three hidden layers with 2, 15, and 2 neurons, respectively. In Figures 6 and 7, the regression plots of Type-I and Type-II models are presented. The

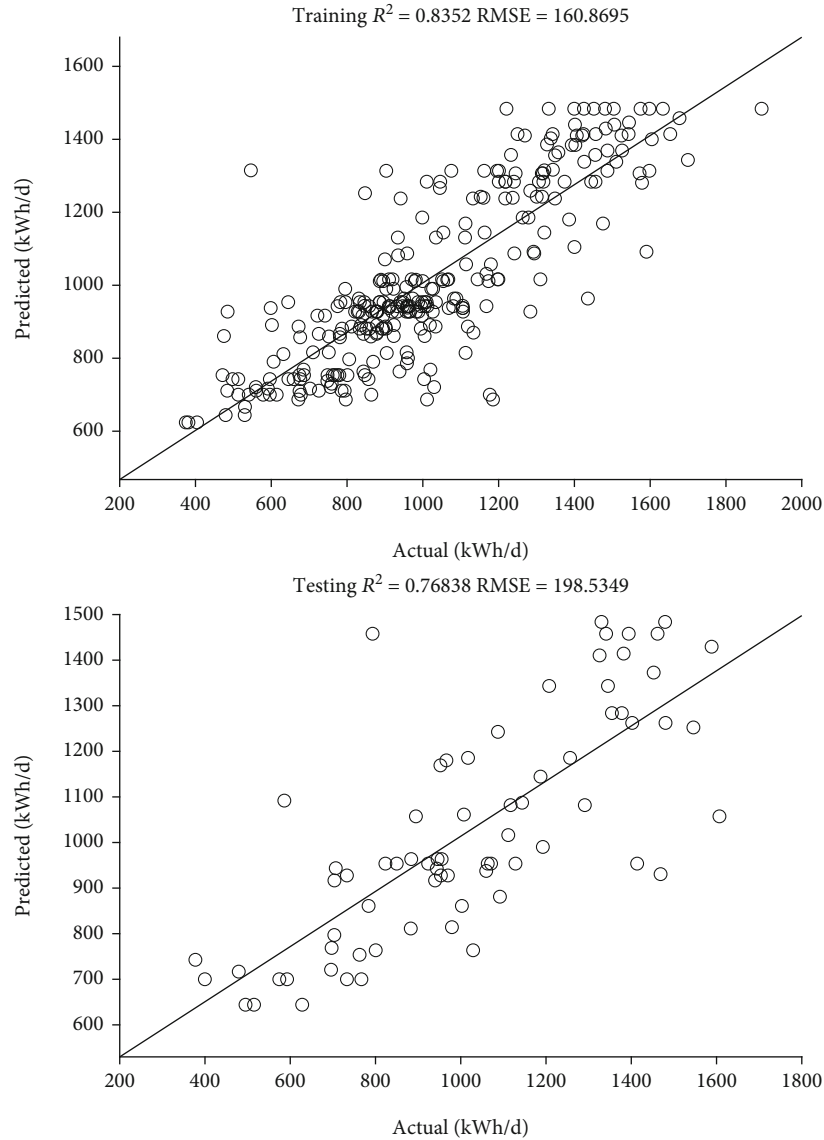


FIGURE 11: Training and testing regression plots for ensemble-II model.

minimum objective function value recorded for Type-I and Type-II model are 10.29 and 10.34, respectively. On the training dataset, we recorded for the Type-I model R^2 and RMSE values of 0.80 and 171.36, respectively. With the testing dataset, the Type-I model yielded R^2 and RMSE values of 0.79 and 187.72, respectively. The Type-II model yielded R^2 and RMSE values of 0.80 and 173.67, respectively, on the training set and R^2 and RMSE values of 0.80 and 185.12, respectively, on the testing set.

4.4. Decision Tree. The hyperparameters of the decision tree algorithm are summarized in Table 6. The optimization of the DT-I model converged to minimum leaf size (MLS) of four and maximum number of splits (MNS) of seven. Conversely, the DT-II model has a higher minimum leaf size of 27 and maximum number of splits of 5. The optimization of the Type-I and Type-II models yielded converged to minimum objective values of 10.41 and 10.4, respectively. In

Figures 8 and 9, the regression plots of Type-I and Type-II models are presented. In both Type-I and Type-II models, the all the features (variables) were sampled. For the Type-I model, R^2 and RMSE values of 0.83 and 161.96 were recorded on the training set, while R^2 and RMSE values of 0.66 and 234.55 were recorded on the testing set. Conversely, the Type-II model yielded R^2 and RMSE values of 0.81 and 170.15, respectively, on the training set. On the testing set, R^2 and RMSE values of 0.75 and 202.87 were recorded, respectively, for the Type-II model.

4.5. Ensemble Learners. Table 7 summarizes the hyperparameters of the ensemble learning algorithms for both Type-I and Type-II models. The optimization algorithm converged to minimum objective function values of 10.34 and 10.36 after 100 iterations. The Type-I model converged to the *Bag* method, while the Type-II model converged to the *LSBoost* method. In Figures 10 and 11, the regression plots

of Type-I and Type-II models are presented. Compared with the case of decision tree algorithm, the Type-I Ensemble learner model uses a subset of the total number of features, that is, the best minimum objective function value in this case was obtained by using just five out of the seven features. The number of learning cycles (NLC) in the Type-II model was higher compared to the Type-I model. In terms of model performances, in the case of the Type-I model, the ensemble learners gave R^2 and RMSE values of 0.89 and 135.11, respectively, on the training set, and 0.80 and 183.89 on the testing set, respectively. In the instance of the Type-II model, R^2 and RMSE values of 0.83 and 160.86, respectively, were recorded on the training set. On the testing set, we obtained R^2 and RMSE values of 0.76 and 198.53, respectively, for the Type-II model.

5. Conclusions

In this study, we have developed different computational models and carried out comparative analysis between the developed models using R^2 and RMSE performance criterion. The summary of the developments in this study is given as follows:

- (i) Five machine learning-based algorithms were employed to develop a day-ahead electricity demand models representing electric vehicles in Barcelona
- (ii) Two main model classes were formulated in this study; the Type-I model was formulated by considering demand from the previous seven days as features, that is $L(k-1) \cdots L(k-7)$; the Type-II model was formulated by considering only two features namely, the consumption of the previous day and the consumption of the same day in the previous week
- (iii) Each of the machine learning algorithms were used to train the proposed model. We compared the performances of Types-I and II models for all algorithms and find that although the Type-II model has less features, it gives almost the same results as the Type-I model, implying that Type-II models has less complexity and computational times

In the case of the Type-I models, the performance of the machine learning algorithms can be summarized from best to worst as follows: $ENSEMBLE > ANN > SVR > GPR > DT$.

For the Type-II models, the performances of the machine learning algorithms can be summarized as follows from best to worst as follows: $SVR > ANN > GPR > ENSEMBLE > DT$.

Data Availability

Data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Interdisciplinary Research Center for Renewable Energy and Power Systems at King Fahd University of Petroleum and Minerals (KFUPM) under Project INRE2221.

References

- [1] F 24, "Eu parliament approves ban on new fossil-fuel cars from 2035," 2022, <https://www.france24.com/en/europe/20220609-eu-parliament-approves-ban-on-new-fossil-fueled-cars-by-2035>.
- [2] "New registrations of electric vehicles in europe," [https://www.eea.europa.eu/ims/new-registrations-of-electric-vehicles#:~:text=The%20recently%20proposed%20legislation%20\(Fit,needed%20to%20achieve%20these%20goals](https://www.eea.europa.eu/ims/new-registrations-of-electric-vehicles#:~:text=The%20recently%20proposed%20legislation%20(Fit,needed%20to%20achieve%20these%20goals).
- [3] N. Moiseev, A. Mikhaylov, I. Varyash, and A. Saqib, "Investigating the relation of GDP per capita and corruption index," *Entrepreneurship and Sustainability Issues*, vol. 8, no. 1, pp. 780–794, 2020.
- [4] V. Mutalimov, I. Kovaleva, A. Mikhaylov, and D. Stepanova, "Assessing regional growth of small business in Russia," *Entrepreneurial Business and Economics Review*, vol. 9, no. 3, pp. 119–133, 2021.
- [5] I. Varyash, A. Mikhaylov, N. Moiseev, and K. Aleshin, "Triple bottom line and corporate social responsibility performance indicators for Russian companies," *Entrepreneurship and Sustainability Issues*, vol. 8, no. 1, pp. 313–329, 2020.
- [6] V. I. Bushukina, "Specific features of renewable energy development in the world and Russia," *Financial Journal*, vol. 13, no. 5, pp. 93–107, 2021.
- [7] N. S. Matveeva, "Legislative regulation financial statement preparation by micro entities: international experience," *Financial Journal*, vol. 13, no. 5, pp. 125–138, 2021.
- [8] G. V. Kriekinge, C. D. Cauwer, N. Sapountzoglou, T. Coosemans, and M. Messagie, "Day-ahead forecast of electric vehicle charging demand with deep neural networks," *World Electric Vehicle Journal*, vol. 12, no. 4, p. 178, 2021.
- [9] M. H. Amini, A. Kargarian, and O. Karabasoglu, "ARIMA-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation," *Electric Power Systems Research*, vol. 140, pp. 378–390, 2016.
- [10] D.-H. Lee, M.-S. Kim, J.-H. Roh, J.-P. Yang, and J.-B. Park, "Forecasting of electric vehicles charging pattern using Bayesians method with the convolution," *IFAC-PapersOnLine*, vol. 52, no. 4, pp. 413–418, 2019.
- [11] Y. Kim and S. Kim, "Forecasting charging demand of electric vehicles using time-series models," *Energies*, vol. 14, no. 5, p. 1487, 2021.
- [12] M. B. Arias and S. Bae, "Electric vehicle charging demand forecasting model based on big data technologies," *Applied Energy*, vol. 183, pp. 327–339, 2016.
- [13] M. Majidpour, C. Qiu, P. Chu, H. R. Pota, and R. Gadh, "Forecasting the EV charging load based on customer profile or

- station measurement?," *Applied Energy*, vol. 163, pp. 134–141, 2016.
- [14] Q. Sun, J. Liu, X. Rong et al., "Charging load forecasting of electric vehicle charging station based on support vector regression," in *2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, Xi'an, October 2016.
- [15] Y. Li, Y. Huang, and M. Zhang, "Short-term load forecasting for electric vehicle charging station based on niche immunity lion algorithm and convolutional neural network," *Energies*, vol. 11, no. 5, p. 1253, 2018.
- [16] N. Andrenacci, F. Karagulian, and A. Genovese, "Modelling charge profiles of electric vehicles based on charges data," *Open Research Europe*, vol. 1, p. 156, 2021.
- [17] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [18] J. Ross Quinlan, *C4.5*, Morgan Kaufmann, Oxford, England, 1992.
- [19] J. R. Quinlan, "Improved use of continuous attributes in c4.5," *Journal of Artificial Intelligence Research* 4, vol. 4, pp. 77–90, 1996.
- [20] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and regression trees*, Chapman ve Hall, Routledge, 2017.
- [21] J. Cao, Z. Lin, G.-B. Huang, and N. Liu, "Voting based extreme learning machine," *Information Sciences*, vol. 185, no. 1, pp. 66–77, 2012.
- [22] Y. Lan, Y. C. Soh, and G.-B. Huang, "Ensemble of online sequential extreme learning machine," *Neurocomputing*, vol. 72, no. 13–15, pp. 3391–3395, 2009.
- [23] J.-J. Cao, S. Kwong, R. Wang, and K. Li, "A weighted voting method using minimum square error based on extreme learning machine," in *2012 International Conference on Machine Learning and Cybernetics*, Xian, July 2012.
- [24] B. E. Boser, I. M. Guyon, and V. N. Vapnik, *A Training Algorithm for Optimal Margin Classifiers*, in: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory - COLT 92*, ACM Press, 1992.
- [25] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943.