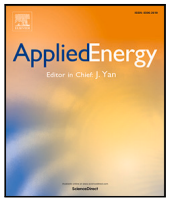
Applied Energy 347 (2023) 121446

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy



Electricity price forecasting in New Zealand: A comparative analysis of statistical and machine learning models with feature selection

Gaurav Kapoor, Nuttanan Wichitaksorn *

Department of Mathematical Sciences, Auckland University of Technology, Private Bag 92006, Auckland 1142, New Zealand

ARTICLE INFO

Keywords:
Electricity price forecasting
GARCH
Stochastic volatility
Machine learning
Feature selection
New Zealand electricity market

ABSTRACT

In this study, we present an empirical comparison of statistical models and machine learning models for daily electricity price forecasting in the New Zealand electricity market. We demonstrate the effectiveness of GARCH and SV models and their t-distribution variants when paired with feature selection techniques, including LASSO, mutual information, and recursive feature elimination. A key aspect of our study is the inclusion of a diverse set of explanatory variables in all models. We compare these models against a range of popular machine learning models, including LSTM, GRU, XGBoost, LEAR, and a four-layer DNN, where the latter two are considered benchmarks. Our results reveal that GARCH and SV models, particularly their t variants, perform exceptionally well when paired with feature selection techniques and explanatory variables. In most scenarios considered, these models outperform machine learning models when coupled with LASSO feature selection. This contribution provides a comprehensive evaluation of the performance of different models and feature selection techniques for electricity price forecasting in the New Zealand electricity market. Our best-performing model improves the symmetric mean absolute percentage error (sMAPE) and mean absolute scaled error (MASE) by 2% to 3% over the LEAR benchmark model, highlighting the practical relevance of our findings.

1. Introduction

1.1. New Zealand electricity market and price forecasting

The New Zealand electricity market (NZEM) is a wholesale market operated by the Electricity Authority (EA), which is the independent regulator of the electricity industry in New Zealand. The EA is responsible for ensuring that the market is fair and efficient and that the market rules are followed. Unlike most European markets, the NZEM follows a real-time market design, where prices are determined every 30 min based on the supply and demand of electricity in that interval. Generators submit offers for the next half-hour period, through a Wholesale Information and Trading System (WITS). The offer consists of a specified quantity of electricity generation at a nominated price. The system operator, Transpower, uses a scheduling, pricing, and dispatch (SPD) method to rank the generation offers in order of price. The SPD method is a merit-order dispatch, where the cheapest generation offers are dispatched first, until all demand for the period is satisfied. The highest-priced bid, offered by a generator, required to meet demand for a given half-hour period is set as the spot price for that trading period.

The NZEM is characterized by a high penetration of renewable energy sources, such as hydro and wind power. In 2021, 82.1% of electricity was generated using renewable sources, see [1]. While this is the right step to reduce the carbon footprint, it also results in a significant increase in the variability of electricity prices, due to the high dependence on suitable climate conditions for hydro and wind power generation. Furthermore, the New Zealand government has a strong focus on achieving 100% renewable electricity generation by 2030. As a result, it becomes increasingly crucial to develop accurate and reliable electricity price forecasting models to help the electricity market participants make informed decisions.

Electricity price forecasting (EPF) has been a topic of interest for more than two decades now, since the deregulation trend of electricity markets around the world. There have been numerous models and methodologies implemented in EPF literature. Literature on EPF models generally falls under one of five categories; game theory models, fundamental models, reduced-form models, statistical models, and machine learning models. In this study, we focus on the comparison between statistical models and machine learning models as they have seen the most effective results. See [2–4] for comprehensive reviews of advances in EPF in the last two decades.

* Corresponding author.
E-mail addresses: gaurav.kapoor@aut.ac.nz (G. Kapoor), nuttanan.wichitaksorn@aut.ac.nz (N. Wichitaksorn).

https://doi.org/10.1016/j.apenergy.2023.121446
Received 13 April 2023; Received in revised form 17 May 2023; Accepted 10 June 2023
Available online 19 June 2023

In this study, we focus on the daily EPF problem for the New Zealand electricity market, which is a real-time market. We focus on daily prices since many of these explanatory features are not available on a lower frequency, and we do not wish to conduct a mixed-frequency study, since the primary focus of this research is the comparison between GARCH, SV, LEAR, and machine learning benchmarks. We forecast prices for the five major regions in New Zealand, the upper North Island (UNI), the central North Island (CNI), the lower North Island (LNI), the upper South Island (USI), and the lower South Island (LSI). These regions encompass the whole of New Zealand and several metrics in the New Zealand electricity market are represented based on these regions. The full dataset consists of prices from 01/01/2014 to 31/10/2022. We apply a rigorous time-series cross-validation scheme when producing forecasts, to obtain robust estimates. A key consideration in our work is the incorporation of numerous explanatory features relating to the New Zealand electricity market. This includes load, generation by fuel type, forward prices from the derivatives market, reserve prices, transmission information, and weather features. Furthermore, the weather features include temperature, precipitation information, humidity, wind information, dew, cloud coverage, and solar energy. We construct a unique set of features for each region mentioned above. To our knowledge, there has not been such a comprehensive study of EPF in the New Zealand real-time market, which is characterized by a high concentration of renewable energy sources.

With a large set of explanatory features, we identify several groups of highly correlated features. To remedy this, we incorporate several feature selection techniques. In particular, we apply the LASSO, mutual information (MI), and recursive feature elimination (RFE) feature selection methods individually on all models described above apart from the LEAR model. We test the performance of each model with each of these methods, as a consistent measure of their performance. One of our key findings suggests that incorporating LASSO with GARCH-t (LE-GARCH-t) improved the GARCH model's predictions by up to 40% over GARCH-t with all features. We see similar results with LE-GARCH, LE-SV-t, LE-SV, and also variants of these models using MI. We also conduct Diebold–Mariano tests, see [5], on the model forecasts. We find the models mentioned to be consistently in the group of top-performers, along with the LEAR benchmark.

1.2. Electricity price forecasting using GARCH and stochastic volatility models

The well-known GARCH model of [6] is well-suited for heteroskedastic time series data. Its ability to model the conditional variance of a time series has made it a popular choice in financial econometrics. However, to no surprise, it has seen many application in EPF literature as well, due to the presence of heteroskedasticity in electricity prices, [7].

[8] provided one of the first applications of GARCH models to EPF. In particular, they successfully implemented an AR-EGARCH specification to model California energy prices during a crisis period in 2000. Generally, the earlier implementations of GARCH models, from approximately mid-2000s to early 2010s, were shown to be effective when compared with other models, for example, see [9,10].

Most implementations of the GARCH specification in EPF involve some form of hybrid setting with other statistical models, most commonly, an AR, ARMA, or ARIMA specification. For example, [7] use a wavelet transform to decompose electricity prices into subseries, and predict them using an ARIMA-GARCH model. [10] present an adaptive wavelet neural network with an ARMAX-GARCH specification for the PJM market. Most recently, [11] compare the performance of an ARIMA-GARCH models along with 26 other models, for predictions in several EPEX markets. Their results suggest that the performance of the ARIMA-GARCH and other traditional statistical models is worse than that of the machine learning models they used in their study. However, importantly, they considered an ARIMA-GARCH model without the implementation of exogenous variables, whereas most of the machine learning models had access to these variables.

While there is vast literature on the use of GARCH models for EPF, only a small percentage of those implement GARCH with exogeneous variables. [10] consider an ARMAX-GARCH model, however they only have a single explanatory variable, which is electricity load. [12] employ a Reg-ARFIMA-GARCH model with several fundamental explanatory variables in the Italian electricity market. [13] study the prediction capabilities of ARIMAX-GARCH and ARIMAX-GARCHX frameworks using weather variables. All of these studies suggest that incorporating exogoneous variables improve model forecasting capabilities.

To our knowledge, newer studies, which provide GARCH specifications as a benchmark for statistical models, have rarely considered GARCH with exogeneous variables or thorough feature selection. This is likely due to the primary focus on machine-learning methods, as well as the notion that introducing exogeneous variables with GARCH is unnecessary since the variables will be able to capture the heteroskedastic behavior of electricity prices, see [14]. However, in this study, we find that the inclusion of exogoneous variables in GARCH models can improve forecasting performance, and that feature selection is important for GARCH models as well. In this study, we implement the GARCH(1,1) model and the GARCH(1,1)-t model. Both variants are estimated using the `ARCH` package in `Python`, see [15], which implements a maximum likelihood estimation technique.

As a comparison for GARCH, we consider a stochastic volatility (SV) framework, where the volatility is specified as a latent stochastic process, as proposed by [16]. They are unlike the GARCH specification, where the evolution of volatility is deterministic. Despite the early evidence in favor of SV models, see [17,18], these models have not found comparable success in the field of time-varying volatility modeling. This is largely due to the estimation difficulty and incompatibility of estimation methods with different SV frameworks, see [19]. As a comparison, the GARCH family has numerous variants, and requires only a few tweaks in the estimation procedure.

SV models have not seen significant implementation or success in EPF, however, they have been applied in financial econometrics settings and commodities markets, see [20]. Furthermore, there are several studies comparing the forecasting capabilities of GARCH and SV models. For example, [21] provide a comprehensive study of forecasting oil, petroleum products and natural gas prices using several GARCH and SV variants. They conclude that the SV models almost always outperform their GARCH counterparts. In another study, [22] model the dynamics of Bitcoin and Litecoin using GARCH and SV models. They also observe that the SV models consistently outperform GARCH models. They also suggest that the t-distribution variants of GARCH and SV models show better results. For these reasons, we include SV models in this study as a comparison for GARCH models. We consider the standard SV model as well as the SV-t variant using the `stochvol` package in R, see [23], which implements a Bayesian Markov chain Monte Carlo (MCMC) estimation technique.

1.3. Popular models in electricity price forecasting literature

In recent years, there have been several successful implementations of linear regression models with numerous input features for EPF. It has been observed that performing regularization techniques, such as the least absolute shrinkage and selection operator (LASSO) of [24], to reduce the feature set, can significantly improve forecasting performance, see [11,25–27]. The LASSO adds a penalty term to the objective function of the regression model, which is proportional to the sum of the absolute values of the coefficients. This encourages the coefficients to be small and some coefficients to be zero, and thus, reduces the number of features used in the model. Such an approach is referred to as the LEAR model in [4]. Due to the success of the LEAR model, and its simplicity in theory, we consider it a benchmark in this study, as suggested by [4].

Our second benchmark model is a four-layer deep neural network (DNN) with two-hidden layers, a natural extension of the traditional multilayer perceptron. DNNs are simple but powerful models that form the basis for other advanced machine learning models. They have seen much use in EPF, and often outperform more advanced machine learning models. For example, [11] provide a thorough study of 27 models, of which 15 are statistical models and 7 are machine learning models. They find the DNN, the long short-term memory (LSTM) model, and the gated recurrent unit (GRU) model, to obtain a predictive accuracy that is better than all other models. Furthermore, of these three, they find that DNN is the best performing. For other application of DNN in EPF, see [28,29].

We also consider two recurrent neural network (RNN) models in this study, particularly the long short-term memory (LSTM) model [30], and the gated recurrent unit (GRU) [31]. They are both capable of learning long-term dependencies in time series data, and are able to capture the temporal structure of the data. Both models have seen successful applications in energy-related literature, see [32–34]. As mentioned previously, [11] find that LSTM, GRU, and DNN are the best performing models. In their study, the LSTM and GRU models are hybrid models combined with a DNN. For applications of LSTM and GRU in EPF, see [35,36].

Finally, we also consider the extreme gradient boosting (XGBoost) model [37]. The XGBoost model is a popular ensemble of regression trees [38], based on the principle of boosting, which is a sequential technique for constructing an optimal combination of weak learners.

In this study, we perform hyperparameter optimization for the DNN, LSTM, GRU, and XGBoost models using the tree-structured Parzen estimator (TPE) algorithm [39]. TPE is a Bayesian optimization algorithm that is able to efficiently search the hyperparameter space. It is a sequential model-based optimization technique that uses a Gaussian process to model the objective function. It is able to efficiently search the hyperparameter space in order to find the optimal hyperparameters in few iterations. We implement the TPE algorithm using the `hyperopt` package in `Python`, see [40].

1.4. Feature selection methods

In this study, we consider three feature selection methods, the LASSO, mutual information (MI), see [41], and recursive feature elimination (RFE), see [42]. Each of these techniques is individually applied to each of the models described above, apart from the LEAR model, since it has already undergone LASSO regularization.

MI is a popular choice for feature selection in EPF. For example, [43] implement an iterative two-stage feature selection technique using MI and correlation analysis for optimal selection of lagged prices. They implement this technique within a DNN to make short-term price prediction in the PJM electricity market. As another example, [44] apply MI to select optimal features from their set of lagged prices, load, and available generation.

Relating to RFE, [45] implement the method on a set of 16 features applicable to retail electricity usage for price and load forecasting. In another study, [46] take a hybrid approach to combining the RFE using a support vector machine (SVM) estimator. They also use MI as a pre-processing technique.

All of these methods have seen some success in EPF, particularly the LASSO. As a result, it will be an interesting comparison to see how these methods perform in conjunction with the models.

1.5. Motivation and contributions

Machine learning models have seen numerous successful implementations in EPF. They are very capable tools for extracting patterns from non-linear time-series. [47] provides a thorough review of machine learning models and methodologies for EPF. However, as [4] points out, comparisons between statistical models and machine learning models have been limited. To be specific, advanced studies have not provided fair comparisons between these categories of models. Advanced machine learning models are often compared with simple statistical models, and without thorough feature engineering, feature selection, or cross-validation.

We consider this to be a crucial gap in literature. In particular, we find that the application of GARCH and SV models as comparisons for novel machine learning techniques in EPF has been limited to their basic forms, or at most, in a hybrid form with ARIMA, or other statistical models. We have found a lack of research works pertaining to GARCH and SV models with exogenous features, which is a key component of our study. Furthermore, when exogenous features are included, they are often not selected or engineered in a rigorous manner. This is in contrast to the extensive feature engineering and selection that is applied to machine learning models. This lack of rigor is abundant in the literature, and is a key motivation for this study.

To resolve this gap in literature, this study provides a fair comparison of the auto-regressive mean with generalized autoregressive conditional heteroskedasticity volatility (GARCH) and auto-regressive mean with stochastic volatility (SV) models and their t-distribution variants with a variety of machine learning models, including the long short-term memory (LSTM) model, the gated recurrent unit (GRU), extreme gradient boosting (XGBoost), the LASSO-estimated auto-regressive (LEAR) model, and a four-layer deep neural-network (DNN).

Following the guidelines of [4], we consider the LEAR and DNN models as benchmarks in this study. The least absolute shrinkage and selection operator (LASSO) as a feature-selection tool has been effective in many applications with a considerable number of explanatory variables, including EPF. Similarly, simple DNNs also provide an effective basis to compare with traditional and more advanced machine learning models. Whether LEAR is a machine learning model or statistical model is a topic for debate, since auto-regressive models are generally statistical models, however regularization is seen as a machine learning technique by some. Regardless of its classification, the model is a consistent benchmark for both categories of models.

We employ the GARCH and SV frameworks, in particular, due to their ability to capture time-varying volatility dynamics, which is abundantly present in electricity prices, see [7]. GARCH models have been applied in several forecasting studies, however, their results suggest they perform no better than simpler AR models, for example, see [11]. In another study, [14] suggest that the effectiveness of GARCH diminishes when the fundamental drivers of electricity price volatility are accounted for using explanatory variables. However, our results contradict this statement. It is true that the GARCH and SV models severely underperform the benchmarks when they are overburdened with explanatory features. However, with a parsimonious set of features, we find that the GARCH and SV models outperform the benchmarks (LEAR and DNN models), on average, by 2% and 3% in terms of the symmetric mean absolute percentage error (sMAPE) and mean absolute scaled error (MASE), respectively.

It is important to discuss a potential shortcoming of our study. In particular, we focus on simpler machine learning models. We primarily do this because past literature, such as [4,11], suggest that simpler base models such as LEAR, DNN, and LSTM, can outperform newer, complex models. However, recent research also speaks of the usefulness of transformers for time-series forecasting, which we have excluded from our study. On the other hand, we also employ fairly simple base statistical models in the form of GARCH and SV. We believe the level of complexity between the statistical and machine learning models in our study is comparable, and thus, provides a fair comparison.

With these contributions, we hope to provide a fair and comprehensive comparison between statistical and machine learning models in EPF.

Table 1
Descriptive statistics of the electricity price data. All values are in NZD/MWh.

	UNI	CNI	LNI	USI	LSI
mean	107.45	102.85	101.29	103.59	95.37
std	73.81	73.51	89.54	76.82	69.85
skew	3.05	3.32	15.36	2.86	2.60
kurtosis	26.92	27.61	505.24	20.10	16.60
min	0.02	0.02	0.02	0.02	0.02
25%	60.91	57.86	56.54	55.05	50.63
50%	84.10	80.27	79.39	80.85	74.11
75%	131.37	124.87	122.10	127.64	117.81
max	1330.57	1237.19	3288.31	1250.05	1092.47

1.6. Paper structure

This paper is structured as follows. Section 1 provides a review of EPF literature pertaining to the models and methodologies employed in this paper. Section 2 introduces the data and discusses the preprocessing and feature selection techniques. Section 3 describes the models used in this paper. Section 4 presents the forecasting results of our study and provides a discussion. Finally, Section 5 concludes the paper.

2. Data and preprocessing techniques

2.1. Price data

The data consists of daily electricity prices across New Zealand from 01/01/2014 to 31/10/2022. In particular, the price data, as well as some of the features, are separated into five regions across New Zealand, the upper North Island (UNI), the central North Island (CNI), the lower North Island (LNI), the upper South Island (USI), and the lower South Island (LSI). We attempt predictions for all five regions in separate univariate frameworks. These regions together encompass the entire New Zealand electricity grid. We have chosen to study these regions, rather than specific nodes, since they provide broader insights into the behavior of electricity prices within each region. Furthermore, several metrics relating to the electricity market, provided officially by the NZ electricity authorities, are separated by these regions. Electricity price and demand vary in each region based on population density and industrial activity. For example, much of the industrial demand for electricity is present in the CNI, whereas the LSI hosts a large amount of generation capacity.

The descriptive statistics of the electricity price data are presented in Table 1. All regions typically behave similarly, however, we observe that LNI has much larger skewness and kurtosis, indicating that this region is much more prone to positive jumps in prices. The LNI region hosts several lakes and is attributed with the highest generation using hydro fuel, so it is not surprising that the region has a higher skewness and kurtosis.

2.2. Features

In this study, we include a variety of features to improve the performance of the models. The time period for all features is the same as that for price data, i.e., from 01/01/2014 to 31/10/2022. Additionally, all features are available at a daily interval. Some of the features are segmented into five individual series, corresponding to the five regions mentioned above. The features under consideration are:

- **Electricity load (MWh)**: Average daily electricity load (demand) for each region under consideration.
- **Generation by fuel type (MWh)**: Average daily generation for each of the following fuels: hydro, wind, coal, gas, geothermal, diesel, and wood. This data is not segmented for each region, it is an average across New Zealand for each fuel type.
- **Reserve prices (NZD/MWh)**: Reserves are generation capacity that is made available to be used in the event of a sudden failure of a generating or transmission facility in order to maintain system frequency at 50 Hz. Fast instantaneous reserve (FIR) is available within six seconds and must be able to operate for one minute. Sustained instantaneous reserve (SIR) is available within 60 s and must be available for 15 min. FIR and SIR prices are segmented into two regions: North Island and South Island.
- **Forward prices (NZD/MWh)**: Forward prices are taken from the New Zealand electricity derivatives market, which is listed under the Australian Securities Exchange (ASX). Forward prices indicate the average price of electricity for the time period under consideration. We consider quarterly contracts, meaning that forward prices are estimates of average daily prices for specific calendar quarters. We also consider base contracts rather than peak contracts, meaning that all 24 h in a day are considered when averaging. The following contract schemes are used as features:
  - **All maturities**: Average settlement price of all contracts currently being traded.
  - **Short-dated maturities**: Average settlement price of contracts maturing within the next 12 months of current date.
  - **Long-dated maturities**: Average settlement price of contracts maturing more than 12 months from current date.
  - **Year+1 maturities**: Average settlement price of contracts maturing in the year preceding current year.
  - **Year+2 maturities**: Average settlement price of contracts maturing in the second year preceding current year.
  - **Year+3 maturities**: Average settlement price of contracts maturing in the third year preceding current year.

  The forward prices are available for two specific nodes in the New Zealand national grid: Otahuhu in the North Island, and Benmore in the South Island. Therefore, each of the features described above are available for each of the two nodes.
- **HVDC transfer (GWh)**: The daily net energy transferred across the HVDC link that connects Benmore in the South Island to Haywards in the North Island. Positive values indicate net northward flow to Haywards, negative values indicate net southward flow to Benmore.
- **Weather data**: We have acquired daily weather data from several weather stations across New Zealand. Each of the variables described below are segmented into the five regions mentioned above. The variables are:
  - **Wind Speed (kph)**: Average daily wind speed.
  - **Wind Direction (degrees)**: Average daily wind direction.
  - **Precipitation (mm)**: Average daily precipitation.
  - **Precipitation Coverage (%)**: Average daily precipitation coverage. Precipitation coverage is the proportion of the day that measurable precipitation occurs.
  - **Temperature (C)**: Average daily temperature.
  - **Dew (C)**: Average daily dew.
  - **Humidity (%)**: Average daily humidity.
  - **Cloud Cover (%)**: Average daily cloud cover. Cloud cover is the proportion of sky near the region that is covered by clouds.
  - **Solar Energy (MJ/m^2)**: Average daily solar energy.

  Again, each of these weather variables is available for each of the five regions under consideration. Additionally, we include average wind speed and wind direction specifically around the wind farms across New Zealand and precipitation and precipitation coverage across hydro dams and lakes across New Zealand as four additional features.

A correlation heatmap of the price data and features is presented in Fig. 1. For clarity purposes, we provide broad labels for all features, with smaller labels indicating the segmented data. We observe that several groups of features are highly correlated with each other. Particularly, the forward prices exhibit multicollinearity, as do some of the weather variables.

2.3. Data preprocessing

The price data and features are all standardized using one of three techniques, depending on original probability distribution. Series, which exhibit heavy-tailed distributions, are first standardized using either the Box–Cox transformation or the Yeo-Johnson transformation, depending on whether the data is strictly positive or not, and are then scaled using min–max scaling. Series, which exhibit close to normal distribution properties, are simply scaled using min–max scaling. As a special case, wind direction, which is originally measured in degrees, is first transformed using a sine transformation, and then scaled using min–max scaling. To further elaborate on the transformations, we find that electricity prices, coal generation, diesel generation, reserve prices, forward prices, and precipitation series tend to display a skewed distribution, with median values being far from the mean. This suggests a higher tendency for extreme values, due to which, we choose to transform them using the Box–Cox or Yeo-Johnson logarithmic transformations. The remaining features typically have symmetric distributions, and it suffices to simply scale them using min–max scaling. The transformations and min–max scaling are defined as follows:

$$\begin{aligned}
\text{Box–Cox: } x' &= \begin{cases} \frac{x^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \ln(x) & \text{if } \lambda = 0 \end{cases} \\
\text{Yeo-Johnson: } x' &= \begin{cases} \frac{(x+1)^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0, x \geq 0 \\ \ln(x+1) & \text{if } \lambda = 0, x \geq 0 \\ -\frac{(-x+1)^{2-\lambda} - 1}{2-\lambda} & \text{if } \lambda \neq 2, x < 0 \\ ln(-x+1) & \text{if } \lambda = 2, x < 0 \end{cases} \\
\text{Min-Max: } x' &= \frac{x - \min(x)}{\max(x) - \min(x)},
\end{aligned} \tag{1}$$

where x is the original data, x' is the transformed data, and λ is the transformation parameter. The transformation parameter is determined using the Box–Cox and Yeo-Johnson transformations, and the min–max scaling range is $[0, 1]$. The transformation methods for each series are summarized in Table 2. Keep in mind that features which are segmented into different regions have different scaling parameter λ for each region. The relevant scaling parameters are estimated through maximum likelihood estimation using the `Scikit-Learn` package in `Python`.

After transforming the data, we add several features to the dataset. We introduce lags of 1, 2, 3, 7, and 14 days for each of the price series, as well as the features. Additionally, we add one-hot encoded features representing the day of the week, as well as a binary dummy variable representing whether the day is a holiday or not. To summarize, the full list of features is as follows:

- Price series for each region, lagged by 1, 2, 3, 7, and 14 days: 25 features.
- Demand series for each region, lagged by 1, 2, 3, 7, and 14 days: 25 features.
- Generation series for each fuel type mentioned above, lagged by 1, 2, 3, 7, and 14 days: 35 features.
- Fast and instantaneous reserve prices for both islands, lagged by 1, 2, 3, 7, and 14 days: 20 features.
- All forward contracts mentioned above for the Benmore and Otahuhu nodes, lagged by 1, 2, 3, 7, and 14 days: 60 features.
- Daily HVDC transfer, lagged by 1, 2, 3, 7, and 14 days: 5 features.
- All weather variables mentioned above for each region, plus an additional region for wind speed, wind direction, precipitation, and precipitation coverage, all lagged by 1, 2, 3, 7, and 14 days: 245 features.
- One-hot encoded features representing the day of the week: 7 features.
- Binary dummy variable representing whether the day is a holiday or not: 1 feature.

In total, the full feature set contains 423 features. The full list of features is used for the feature selection process described later in this section. To make the point clear, we strictly use lagged variables for prediction at time t, and we do not use any information from the future.

Table 2
Transformation methods for price and features.

Series	Transformation
Price	Box–Cox + Min-Max
Demand	Min-Max
Hydro Generation	Min-Max
Wind Generation	Min-Max
Coal Generation	Yeo-Johnson + Min-Max
Gas Generation	Min-Max
Geothermal Generation	Min-Max
Diesel Generation	Yeo-Johnson + Min-Max
Wood Generation	Min-Max
Reserve Prices	Yeo-Johnson + Min-Max
Forward Prices	Box–Cox + Min-Max
HVDC Transfer	Min-Max
Wind Speed	Min-Max
Wind Direction	Sine + Min-Max
Precipitation	Yeo-Johnson + Min-Max
Precipitation Coverage	Min-Max
Temperature	Min-Max
Dew	Min-Max
Humidity	Min-Max
Cloud Coverage	Min-Max
Solar Energy	Min-Max

2.4. Cross-validation scheme

Prior to feature selection, we perform a 5-fold time-series cross-validation scheme to the split the data into 5 subsets of training and test sets. Since we are working with time-series data, it is crucial to ensure that the sequential nature of the data is preserved during the cross-validation process. Generally, in time-series cross-validation, the training set contains all available information prior to the test set. However, in this study, we have fixed the length of the training set for each fold to be the same. To be precise, each training set is 1456 days long, and each test set consists of the 364 days immediately succeeding its training set. These values are chosen so that the complete dataset is split into 5 folds of equal length. Fig. 2 illustrates the cross-validation scheme.

The training set for each fold is used to train the model, and the test set is used to evaluate the model. However, in certain cases, we require a validation set to tune hyperparameters. In this case, we split the training set into a training set and a validation set. To be specific, we leave the first 1092 days in the training set, and use the remaining 364 days as the validation set. We utilize a validation set for tuning hyperparameters during feature selection and training with machine learning models.

2.5. Feature selection

We perform feature selection on each cross-validation fold separately. This is because we want to ensure that the features selected for each fold are not dependent on any data from the test sets. We use validation sets to select optimal hyperparameters for each feature selection method.

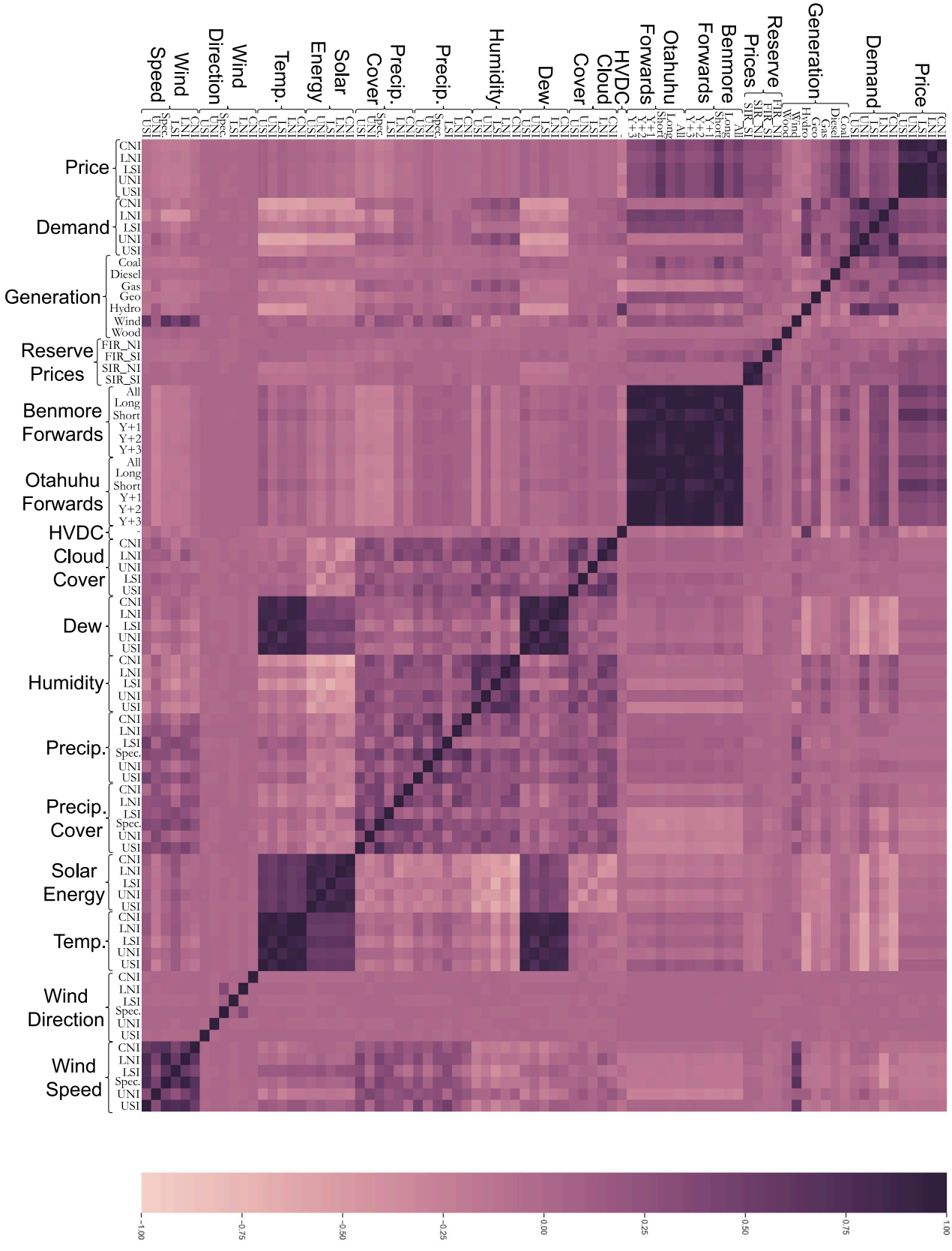


Fig. 1. Correlation heatmap of price and features.

Fig. 2. Cross-validation scheme for model training and testing.

2.5.1. LASSO regularization

LASSO regression is a type of linear regression that is popular for feature selection and regularization. The LASSO adds a penalty term to the ordinary least squares cost function in order to reduce the number of features used in the model. Consider a linear regression model with p features, $x_1, x_2, \ldots, x_p$, and a target variable y

$$y_t = \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \beta_p x_{p,t} + \epsilon_t, \tag{2}$$

the LASSO estimator for the regression coefficients is given by

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{t=1}^{T} (y_t - \beta_1 x_{1,t} - \beta_2 x_{2,t} - \cdots - \beta_p x_{p,t})^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}. \tag{3}$$

This is essentially the LEAR model when we consider the features to have autoregressive components, as is the case with our features described in Section 2.3.

The regularization parameter λ controls the degree to which the LASSO estimator shrinks the regression coefficients towards zero. The larger the value of λ, the more coefficients are shrunk towards zero. In this study, we use a validation set to select the optimal value of λ. For each cross-validation set, we perform a grid search over the values of λ in the range $[10^{-4}, 10^{-1}]$ with a step size of 10^{-3}. For each λ value, we assess the performance of the model on the validation set using the mean squared error (MSE) metric. The value of λ that minimizes the MSE is selected as the optimal value for the corresponding cross-validation fold.

2.5.2. Mutual information

For two random variables, X and Y, the mutual information between X and Y is defined as

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}. \tag{4}$$

The mutual information (MI) between two random variables is a measure of the amount of information that X provides about Y. In this study, we use the MI between the target variable and each feature to select the most relevant features. For feature selection, MI requires us to provide the number of features to select, k. For each cross-validation set, we perform a grid search over the values of k in the range $[1, 423]$. For each k value, we estimate a linear regression model using the training set and assess the performance of the model on the validation set using the MSE metric. The value of k that minimizes the MSE is selected as the optimal value for the corresponding cross-validation fold.

2.5.3. Recursive feature elimination

Recursive feature elimination (RFE) is a feature selection method that recursively removes features from the feature set. The method starts with all features in the feature set, and iteratively removes the least important features. The importance of each feature is determined by a given estimator. To maintain consistency with the other feature selection methods, we use a linear regression estimator to determine the importance of each feature. As with MI, RFE requires us to provide the number of features to select, k. For each cross-validation set, we perform a grid search over the values of k in the range $[1, 423]$. For each k value, we estimate a linear regression model using the training set and assess the performance of the model on the validation set using the MSE metric. The value of k that minimizes the MSE is selected as the optimal value for the corresponding cross-validation fold.

3. Models

3.1. LEAR model

The LASSO estimated auto-regressive (LEAR) model was first implemented in [25] under the name LassoX. It is essentially a linear regression model with a large number of features, where the coefficients are estimated using LASSO regularization. The model has the following specification

$$y_t = \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \beta_n x_{n,t} + \epsilon_t, \epsilon_t \quad \sim \mathcal{N}(0, \sigma^2). \tag{5}$$

where y_t is the electricity price at time t, $x_{i,t}$ is the ith feature at time t, and ϵ_t is the error term at time t. The parameters $\sigma^2, \beta_1, \ldots, \beta_n$ are to be estimated.

The LASSO estimator for the regression coefficients is given by

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{t=1}^{T} (y_t - \beta_1 x_{1,t} - \beta_2 x_{2,t} - \cdots - \beta_n x_{n,t})^2 + \lambda \sum_{i=1}^{n} |\beta_i| \right\}. \tag{6}$$

Following the guidelines of [4], we consider the LEAR model as a benchmark in this study.

3.2. GARCH models

We consider a GARCH(1,1) with exogeneous variables in the mean equation, which has the following specification

$$\begin{aligned} y_t &= \mu_t + \epsilon_t, \\ \mu_t &= \mu_0 + \sum_{i=1}^{n} \phi_i x_{i,t}, \\ \epsilon_t &\sim \mathcal{N}(0, \sigma_t^2), \\ \sigma_t^2 &= \omega + \alpha \epsilon_{t-1}^2 + \beta \sigma_{t-1}^2, \end{aligned} \tag{7}$$

where y_t is the electricity price at time t, μ_t is the mean of the electricity price at time t, and ϵ_t is the error term at time t. The mean equation is dependent on the variables $x_{i,t}$, $i = 1, \ldots, n$, which are the features we employ in this study, see Section 2.3. The variance equation is dependent on the previous error term ϵ_{t-1} and the lagged variance σ_{t-1}^2. The parameters ω, α, β, μ_0, and ϕ_i are to be estimated.

The features $x_{i,t}$, $i = 1, \ldots, n$ differ based on which feature selection algorithm is employed. Using LASSO feature selection results in the same feature set utilized in the LEAR model. We denote this the LASSO-estimated GARCH (LE-GARCH) model. Similarly when using the MI or RFE generated features, we denote the model as MI-GARCH and RFE-GARCH, respectively. When using all features, the model is simply denoted as the GARCH model.

In the case of the GARCH-t model, the error term ϵ_t is distributed as a t-distribution with ν degrees of freedom, $\epsilon_t \sim t_\nu(0, \sigma^2)$. Similar to the GARCH, the GARCH-t model is denoted as LE-GARCH-t, MI-GARCH-t, and RFE-GARCH-t when using the LASSO, MI, and RFE feature selection algorithms, respectively.

We use the `ARCH` package in `Python` to estimate the GARCH models, see [15]. The package uses maximum likelihood estimation (MLE) to estimate the parameters of the GARCH models.

3.3. SV models

We consider a SV model with exogeneous variables in the mean equation, which has the following specification

$$\begin{aligned} y_t &= \mu_t + \epsilon_t, \\ \mu_t &= \mu_0 + \sum_{i=1}^{n} \phi_i x_{i,t}, \\ \epsilon_t &\sim \mathcal{N}(0, e^{h_t}), \\ h_t &= \omega + \alpha(h_{t-1} - \omega) + \epsilon_t^h, \\ \epsilon_t^h &\sim \mathcal{N}(0, \sigma_h^2), \end{aligned} \tag{8}$$

where y_t is the electricity price at time t, μ_t is the mean of the electricity price at time t and ϵ_t is the error term at time t. The mean equation is dependent on the features $x_{i,t}$, $i = 1, \dots, n$. The log-volatility h_t is dependent on the previous log-volatility h_{t-1} and the error term ϵ_t^h. The parameters ω, α, μ_0, ϕ_i, and σ_h^2 are to be estimated.

Similar to the GARCH models, the features $x_{i,t}$, $i = 1, \dots, n$ differ based on which feature selection algorithm is employed. When utilizing the LASSO-estimated feature set, we denote the model as the LASSO-estimated SV (LE-SV) model. Similarly when using the MI or RFE generated features, we denote the model as MI-SV and RFE-SV, respectively. When using all features, the model is simply denoted as the SV model.

In the case of the SV-t model, the error term ϵ_t is distributed as a t-distribution with ν degrees of freedom, $\epsilon_t \sim t_\nu(0, e^{h_t})$. Similar to the SV, the SV-t model is denoted as LE-SV-t, MI-SV-t, and RFE-SV-t when using the LASSO, MI, and RFE feature selection algorithms, respectively.

We use the `stochvol` package in R to estimate the SV models, see [23]. The package estimates SV parameters via Bayesian Markov chain Monte Carlo (MCMC) sampling.

3.4. DNN model

We consider a deep neural network (DNN) with two hidden layers in this study. The network takes as input the features $x_{i,t}$, $i = 1, \dots, n$, and outputs a price prediction y_t. The two hidden layers have n_1 and n_2 neurons, respectively. The activation function for the hidden layers is the rectified linear unit (ReLU) function. The network is trained using the Adam optimizer [48]. We train the model for 100 epochs using a batch size of 64. Furthermore, we utilize a validation set to implement early stopping. Fig. 3 shows the network architecture. When a specific feature selection algorithm is used, we denote the model as the LE-DNN, MI-DNN, and RFE-DNN models, referring to the LASSO, mutual information, and recursive feature elimination algorithms, respectively. When all features are used, the model is simply denoted as the DNN model.

We use the tree-based Parzen estimator (TPE) algorithm to optimize several hyperparameters of the DNN model. The hyperparameters are the number of neurons in the first and second hidden layers, n_1 and n_2, respectively, the learning rate of the Adam optimizer, the dropout rate after each fully-connected layer, and whether or not to use batch normalization. The hyperparameters are optimized using a training set, and the model is evaluated on a validation set, which is independent of the test set used for predictions. The optimized hyperparameters are shown in Table 3.

3.5. LSTM and GRU models

To extend the neural network architecture to account for the temporal nature of the electricity price, we consider recurrent neural networks (RNNs) in the form of the LSTM and GRU models. In our study, both model implementations are fairly simple and have only one hidden layer. The LSTM and GRU models have the same architecture as the DNN model, except that the hidden layer is recurrent. The activation function for these models is the Tanh function, and the recurrent activation is the sigmoid function. Both models utilize the Adam optimizer and early stopping using a validation set. Both models are trained for 100 epochs with a batch size of 64, same as the DNN model.

When a specific feature selection algorithm is used, we denote the LSTM model as the LE-LSTM, MI-LSTM, and RFE-LSTM models, referring to the LASSO, mutual information, and recursive feature elimination algorithms, respectively. When all features are used, the model is simply denoted as the LSTM model. Similarly for the GRU model, the LE-GRU, MI-GRU, and RFE-GRU models are denoted for the respective features, and when all features are used, the model is simply denoted as the GRU model.

The hyperparameters of the LSTM and GRU models are optimized using the TPE algorithm. The hyperparameters are the number of neurons in the hidden layer, n_{LSTM} and n_{GRU}, respectively, and the number of previous time steps used as input to the model. The optimized hyperparameters are shown in Table 3.

3.6. XGBoost model

The extreme gradient boosting (XGBoost) model is a tree-based model that uses gradient boosting to produce a prediction. We optimize

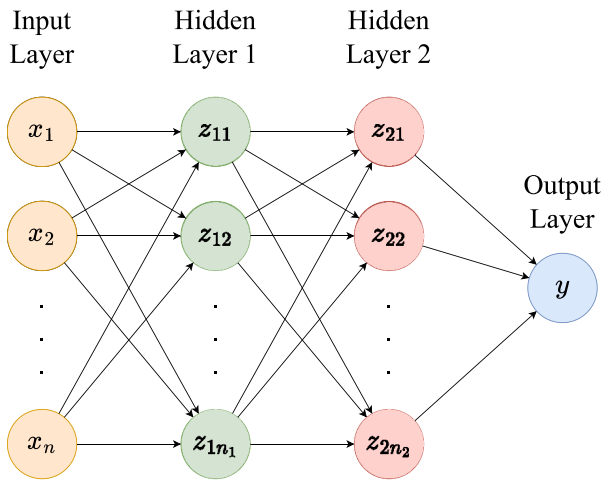


Fig. 3. Network architecture of the DNN model.

Table 3
Optimal hyperparameters of the DNN, LSTM, and GRU models.

Model	Hyperparameter	Value
DNN	n_1	128
	n_2	64
	Learning rate	0.001
	Dropout rate	0.2
	Batch normalization	True
LSTM	n_{LSTM}	80
	Number of time steps	7
GRU	n_{GRU}	96
	Number of time steps	7

Table 4
Optimal hyperparameters of the XGBoost model.

Hyperparameter	Value
$n_{\text{estimators}}$	1000
η	0.1
$n_{\text{max\_depth}}$	5
subsample	0.8
γ	0.1
min_child_weight	1
α	0.1

Table 5
Forecast metrics for North Island regions. *Note*: Bold values indicate best performance for given metric and region.

Model	Central North Island					Upper North Island					Lower North Island				
	MAE	RMSE	MAPE	sMAPE	MASE	MAE	RMSE	MAPE	sMAPE	MASE	MAE	RMSE	MAPE	sMAPE	MASE
DNN	0.4540	0.4719	1.9867	0.7801	1.5015	0.4385	0.4	2.3694	0.7462	1.4734	0.4681	0.4695	2.3892	0.8849	1.5882
GARCH	0.6624	1.115	3.7286	0.8664	2.2058	0.6042	0.8862	4.5744	0.8882	2.0484	0.6296	1.0056	12.2467	0.8828	2.1516
GARCH-*t*	0.6068	0.9114	3.6764	0.8539	2.0248	0.6165	0.8738	3.5651	0.9	2.0692	0.5719	0.8504	7.7705	0.8862	1.964
GRU	0.4202	0.3824	2.5842	0.7263	1.3902	0.4068	0.3443	2.9169	0.7055	1.3708	0.4291	0.4119	3.6281	0.738	1.4538
LEAR	0.3909	0.3556	2.3845	0.686	1.3041	0.3832	**0.3217**	2.9951	0.6822	1.2954	0.3854	**0.3564**	3.1294	0.6965	1.3085
LSTM	0.4536	0.4124	3.2971	0.7492	1.5186	0.4141	0.3587	2.3933	0.7187	1.3966	0.4136	0.389	3.1208	0.7303	1.3996
SV	0.5431	0.7275	2.8784	0.8265	1.8226	0.5428	0.7044	3.4758	0.8299	1.8471	0.5429	0.7232	10.0983	0.8512	1.8549
SV-*t*	0.5431	0.7281	2.9107	0.8353	1.8237	0.5411	0.689	3.6033	0.831	1.8405	0.5416	0.7213	10.1176	0.8475	1.8502
XGBoost	0.4815	0.5239	2.4817	0.7408	1.6434	0.4653	0.4792	3.8907	0.7589	1.6018	0.5787	0.772	4.1283	0.8215	2.0038
LE-DNN	0.447	0.4437	2.0859	0.7852	1.4928	0.5015	0.5047	**2.0238**	0.9041	1.7293	0.4594	0.4828	2.787	0.7987	1.5553
LE-GARCH	0.3976	0.3609	2.2955	0.6943	1.3218	0.3879	0.3337	3.0548	0.6868	1.3086	0.3901	0.3775	2.8878	0.6937	1.3199
LE-GARCH-*t*	**0.3793**	**0.3508**	1.8658	**0.6704**	**1.2626**	**0.3796**	0.3228	2.508	0.6808	**1.2784**	**0.3817**	0.3711	2.6236	**0.688**	**1.2909**
LE-GRU	0.4212	0.386	2.5103	0.7312	1.3991	0.4132	0.3603	2.6946	0.7147	1.3919	0.4202	0.401	3.4032	0.7328	1.4173
LE-LSTM	0.4188	0.3931	2.3699	0.7249	1.3947	0.411	0.3652	2.2914	0.7274	1.3882	0.4114	0.3951	3.0611	0.7425	1.3845
LE-SV	0.3816	0.3537	1.9814	0.6728	1.2694	0.38	0.3228	2.6005	0.6768	1.2806	0.3842	0.3727	2.6887	0.6929	1.2997
LE-SV-*t*	0.3824	0.3548	1.9055	0.6745	1.2718	0.3797	0.3224	2.5207	**0.6754**	1.2787	0.3833	0.3705	2.6317	0.6945	1.2971
LE-XGBoost	0.4418	0.4759	2.1669	0.7273	1.48	0.4263	0.4104	3.1469	0.7287	1.4468	0.492	0.5574	3.2254	0.7812	1.6936
MI-DNN	0.5088	0.5463	**1.8311**	0.8821	1.7157	0.4865	0.4827	2.1955	0.8716	1.6582	0.4892	0.5904	**2.1365**	0.8321	1.6624
MI-GARCH	0.411	0.4035	1.9141	0.723	1.3658	0.3994	0.3534	2.1815	0.7245	1.3527	0.4367	0.4955	5.494	0.7411	1.4627
MI-GARCH-*t*	0.4221	0.4157	1.895	0.7385	1.4022	0.4111	0.3652	2.1956	0.7545	1.3918	0.4105	0.4154	3.7797	0.7424	1.382
MI-GRU	0.4194	0.3794	2.79	0.7268	1.4008	0.431	0.3808	3.1534	0.7209	1.4624	0.4125	0.3946	3.5536	0.7202	1.4042
MI-LSTM	0.416	0.3942	2.5193	0.7192	1.3954	0.411	0.3567	2.4853	0.7135	1.3896	0.4165	0.4053	3.0756	0.7436	1.4172
MI-SV	0.4154	0.4296	1.8261	0.726	1.3806	0.4037	0.3926	2.2591	0.7228	1.3667	0.4002	0.3923	3.2356	0.7256	1.3475
MI-SV-*t*	0.4157	0.4295	1.8191	0.7285	1.3802	0.4035	0.3919	2.2488	0.7231	1.3656	0.4012	0.3931	3.2443	0.7295	1.3506
MI-XGBoost	0.4669	0.499	2.1717	0.7631	1.5727	0.4831	0.4844	4.2121	0.7773	1.661	0.5174	0.6166	3.3341	0.8073	1.7835
RFE-DNN	0.532	0.5926	3.3314	0.8114	1.7617	0.626	0.8712	3.3333	0.9422	2.053	0.6156	1.4859	27.6257	0.8714	2.0741
RFE-GARCH	0.4688	0.5291	2.2	0.8277	1.5676	0.6261	1.1165	4.0893	0.8585	2.0507	0.6587	0.9239	5.2044	0.9508	2.2683
RFE-GARCH-*t*	0.5028	0.5359	2.9153	0.871	1.7081	0.5921	0.9435	3.504	0.8737	1.952	0.5944	0.7748	4.139	0.9972	2.0519
RFE-GRU	0.4395	0.42	3.143	0.7287	1.4724	0.6217	1.0008	3.6674	0.8622	2.0286	0.5152	0.5947	9.3575	0.9156	1.7536
RFE-LSTM	0.4363	0.4199	2.9704	0.7291	1.4646	0.6027	0.9343	3.6783	0.8559	1.9701	0.5069	0.5765	8.0118	0.9014	1.7346
RFE-SV	0.4541	0.4667	2.4743	0.759	1.5175	0.5815	0.9054	3.3983	0.8614	1.9117	0.5008	0.605	6.4682	0.8931	1.7043
RFE-SV-*t*	0.4527	0.4658	2.3219	0.755	1.5128	0.5828	0.9058	3.5744	0.861	1.9154	0.5043	0.6134	6.6761	0.8974	1.7159
RFE-XGBoost	0.5195	0.6091	3.8196	0.7921	1.7418	0.7043	1.2079	6.435	0.9105	2.3318	0.6786	0.9425	7.0961	0.8969	2.3484

the hyperparameters of the XGBoost model using the TPE algorithm. The hyperparameters to optimize are the number of boosting iterations, $n_{\text{estimators}}$, the learning rate, η, the maximum depth of the trees, $n_{\text{max\_depth}}$, the subsample ratio of the training instances, subsample, the minimum loss reduction required to make a further partition on a leaf node of the tree, γ, the minimum sum of instance weight (hessian) needed in a child, min_child_weight, and the L1 regularization term on weights, α. The optimized hyperparameters are shown in Table 4.

When a specific feature selection algorithm is used, we denote the model as the LE-XGBoost, MI-XGBoost, and RFE-XGBoost models, referring to the LASSO, mutual information, and recursive feature elimination algorithms, respectively. When all features are used, the model is simply denoted as the XGBoost model.

4. Results

In this section, we present the forecast metrics and the Diebold–Mariano test results for the models under consideration. The forecast metrics include mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE), symmetric mean absolute percentage error (sMAPE), and mean absolute scaled error (MASE). We also report the Diebold–Mariano (DM) test results to determine whether the improvement in forecast accuracy of one model over another is statistically significant.

4.1. Evaluation metrics

We provide a brief discussion of the metrics used for evaluation of model performance. The three most popular metrics used in EPF research are the MAE, the RMSE, and the MAPE:

$$\text{MAE} = \frac{1}{T}\sum_{t=1}^{T}|y_t - \hat{y}_t| \tag{9}$$

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2} \tag{10}$$

$$\text{MAPE} = \frac{100}{T}\sum_{t=1}^{T}\frac{|y_t - \hat{y}_t|}{y_t} \tag{11}$$

where y_t is the actual value and $\hat{y}_t$ is the forecasted value of the tth observation. The MAE and RMSE are symmetric metrics, while the MAPE is asymmetric. The former two are more sensitive to large errors, while the MAPE is more sensitive to small errors and outliers, and is also not defined for observations with zero actual values. To alleviate some of these issues, we also consider the symmetric mean absolute percentage error (sMAPE):

$$\text{sMAPE} = \frac{100}{T}\sum_{t=1}^{T}\frac{|y_t - \hat{y}_t|}{|y_t| + |\hat{y}_t|} \tag{12}$$

Due to its ease of interpretation, several studies have suggested the use of the mean absolute scaled error (MASE) as a metric for EPF, see [2,49,50]. The MASE is defined as:

$$\text{MASE} = \frac{1}{T}\sum_{t=1}^{T}\frac{|y_t - \hat{y}_t|}{\frac{1}{T-1}\sum_{i=1}^{T}|y_i - y_{i-1}|}, \tag{13}$$

where the denominator inside the outer sum represents the in-sample MAE of a naive one-step ahead forecast. The MASE provides a simple interpretation of being better/worse than a one-step ahead naive forecast if it is lower/higher than one. Furthermore, since the MASE uses the MAE of the naive one-step ahead forecast as a benchmark for accuracy, it is less sensitive to outliers than the other metrics, and provides a more stable baseline for comparison. This is especially important when comparing the performance of EPF models, since electricity prices are prone to large spikes and outliers.

Table 6
Forecast metrics for South Island regions. *Note*: Bold values indicate best performance for given metric and region.

Model	Upper South Island					Lower South Island				
	MAE	RMSE	MAPE	sMAPE	MASE	MAE	RMSE	MAPE	sMAPE	MASE
DNN	0.4861	0.4942	2.3833	0.7975	1.8321	0.4669	0.447	2.4921	0.7819	1.7794
GARCH	0.531	0.6378	3.6149	0.7908	2.0412	0.4904	0.5111	4.139	0.7922	1.9266
GARCH-*t*	0.5451	0.6346	3.403	0.849	2.0639	0.4979	0.5684	3.3643	0.7954	1.9636
GRU	0.3923	0.3178	2.4352	0.6796	1.4842	0.3821	0.3089	2.8662	0.6655	1.4785
LEAR	0.3559	0.2839	2.3291	0.6454	1.3473	**0.3449**	**0.2692**	2.4572	**0.6283**	**1.3365**
LSTM	0.3999	0.3368	2.5603	0.6863	1.5137	0.3979	0.3326	3.269	0.688	1.5381
SV	0.4853	0.5241	2.801	0.7797	1.8558	0.4836	0.5049	3.8839	0.7727	1.8931
SV-*t*	0.4804	0.5175	2.8234	0.7682	1.835	0.4833	0.5072	3.8267	0.7759	1.8913
XGBoost	0.5119	0.5273	3.145	0.7611	2.0024	0.4132	0.3758	3.7167	0.7278	1.6303
LE-DNN	0.435	0.3917	2.3977	0.7528	1.6492	0.4636	0.4575	3.4647	0.7632	1.7794
LE-GARCH	0.3532	**0.2814**	2.3012	**0.6393**	1.3346	0.3525	0.2875	2.8456	0.6427	1.369
LE-GARCH-*t*	**0.352**	0.2858	2.1612	0.6396	**1.328**	0.3486	0.2761	2.7815	0.6402	1.3513
LE-GRU	0.3765	0.3052	2.5336	0.6689	1.4214	0.3984	0.3195	3.5424	0.7009	1.5303
LE-LSTM	0.3879	0.3254	2.4816	0.6927	1.4625	0.3816	0.3058	2.8212	0.6923	1.4685
LE-SV	0.3541	0.285	2.2142	0.6459	1.3357	0.3492	0.2746	2.8589	0.6401	1.3533
LE-SV-*t*	0.3538	0.2857	2.1933	0.645	1.3343	0.348	0.2735	2.7157	0.6346	1.3487
LE-XGBoost	0.4234	0.3979	2.9577	0.7078	1.626	0.4078	0.3578	3.7302	0.6956	1.6069
MI-DNN	0.4258	0.4197	2.1775	0.7261	1.6095	0.4475	0.4216	3.521	0.7873	1.7825
MI-GARCH	0.3598	0.2849	2.244	0.651	1.3649	0.3618	0.2916	2.7337	0.6582	1.4012
MI-GARCH-*t*	0.3645	0.2906	**2.1138**	0.6773	1.385	0.3619	0.2927	2.5224	0.6634	1.4051
MI-GRU	0.4002	0.327	2.7019	0.6998	1.5212	0.3871	0.3096	3.1054	0.6788	1.5027
MI-LSTM	0.3917	0.316	2.615	0.6965	1.4909	0.3744	0.3	2.9024	0.6696	1.4653
MI-SV	0.3637	0.29	2.1543	0.6646	1.3806	0.3603	0.2902	2.6503	0.6534	1.3959
MI-SV-*t*	0.363	0.2904	2.1255	0.6638	1.3779	0.3582	0.2888	**2.3892**	0.6478	1.3885
MI-XGBoost	0.421	0.3883	2.3878	0.7126	1.6104	0.4447	0.4325	2.659	0.7513	1.7422
RFE-DNN	0.5547	0.6397	2.9509	0.8356	2.0991	0.5629	0.6129	2.6783	0.995	2.2083
RFE-GARCH	0.4657	0.5197	2.3796	0.8076	1.7875	0.6296	1.2122	3.3459	0.8851	2.5001
RFE-GARCH-*t*	0.4616	0.5193	2.2769	0.7959	1.7723	0.6234	1.1379	3.1634	0.8895	2.4694
RFE-GRU	0.4472	0.4775	2.2395	0.7667	1.7082	0.5876	0.9465	3.4272	0.8627	2.3161
RFE-LSTM	0.4499	0.4694	2.4281	0.7489	1.7186	0.5935	0.9398	3.2453	0.8674	2.3429
RFE-SV	0.4633	0.5411	2.2794	0.7764	1.7744	0.4733	0.4925	3.0263	0.8407	1.8494
RFE-SV-*t*	0.4635	0.546	2.2716	0.7721	1.7759	0.4715	0.4911	2.7034	0.8354	1.8417
RFE-XGBoost	0.6009	0.7026	3.4217	0.8971	2.358	0.597	0.7286	3.9014	0.951	2.4022

Table 7
Best model for each metric in each region.

	MAE	RMSE	MAPE	sMAPE	MASE
Central North Island	LE-GARCH-*t*	LE-GARCH-*t*	MI-DNN	LE-GARCH-*t*	LE-GARCH-*t*
Upper North Island	LE-GARCH-*t*	LEAR	LE-DNN	LE-SV-*t*	LE-GARCH-*t*
Lower North Island	LE-GARCH-*t*	LEAR	MI-DNN	LE-GARCH-*t*	LE-GARCH-*t*
Upper South Island	LE-GARCH-*t*	LE-GARCH	MI-GARCH-*t*	LE-GARCH	LE-GARCH-*t*
Lower South Island	LEAR	LEAR	MI-SV-*t*	LEAR	LEAR

Table 8
Top three models in each region according to their MASE.

	Best model	Second best model	Third best model
Central North Island	LE-GARCH-*t*	LE-SV-*t*	LE-SV
Upper North Island	LE-GARCH-*t*	LE-SV	LE-SV-*t*
Lower North Island	LE-GARCH-*t*	LE-SV-*t*	LE-SV
Upper South Island	LE-GARCH-*t*	LE-SV-*t*	LE-GARCH
Lower South Island	LEAR	LE-SV-*t*	LE-GARCH-*t*

The evaluation metrics for model forecasts are presented in Tables 5 and 6, for the North Island and South Island regions, respectively. The results shown in these tables are averaged results from the 5-fold cross-validation scheme.

We can make several observations from these tables. Firstly, the LE-GARCH-*t* has the best performance in terms of the MAE, and often in terms of other metrics as well, in all regions except for the LSI region. Other well-performing statistical models include the LE-GARCH, LE-SV-*t*, MI-GARCH-*t*, and MI-SV-t. These models tend to outperform the other machine learning models, and often outperform the DNN and LEAR benchmark models as well. Secondly, we notice that the LEAR model outperforms the GARCH and SV models in the LSI region. LSI prices are generally lower than other regions, and also exhibit lower volatility and price spiking tendencies. Therefore, the outperformance may be due to the GARCH and SV models overfitting in this region. Thirdly, the GARCH, GARCH-*t*, SV, and SV-*t* models with all features included are generally the worst-performing models. This speaks to the importance of regularization or feature selection to construct a parsimonious set of features for these models. In some scenarios, we can observe performance improvement of up to 45% from the GARCH model with all features included to the LE-GARCH or MI-GARCH models. Finally, we observe that the LASSO feature selection generally obtains the best performance metrics, followed by MI. RFE tends to underperform in most cases, but still usually outperforms GARCH and SV models with all features. Table 7 summarizes the main results by displaying the best model according to each metric, for each region.

Table 8 shows the top three best performing models in each region according to the MASE metric. The LE-GARCH-*t* and LE-SV-*t* are in the top three in every region, with the former being the best model in terms of MASE for every region, except the LSI. The LEAR is the best model in the LSI region, and only appears in the top three in this region.

4.2. The Diebold–Mariano test

We utilize the Diebold–Mariano (DM) test, see [5], to assess the statistical significant of our forecast results. We use absolute errors to form the loss differential series:

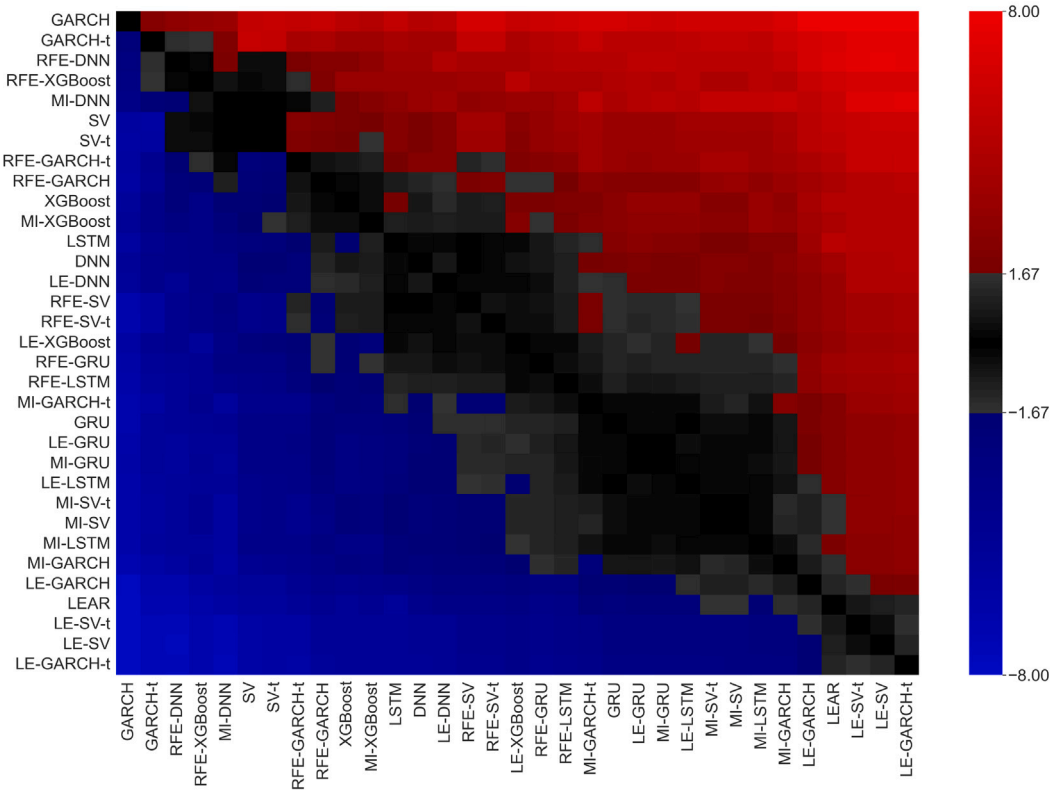


Fig. 4. DM test statistics for Central North Island.

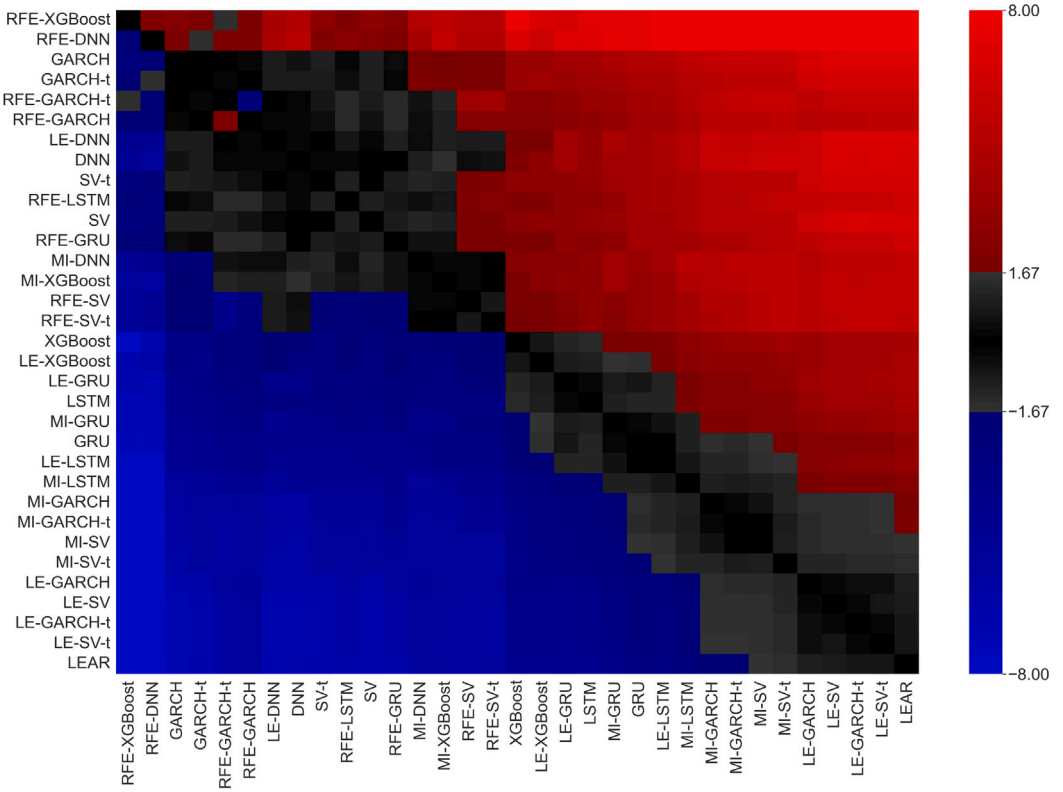


Fig. 5. DM test statistics for Lower South Island.

$$\Delta_t = |\epsilon_{t,A}| - |\epsilon_{t,B}|, \tag{14}$$

where $\epsilon_{t,A}$ and $\epsilon_{t,B}$ are the forecast errors of models A and B at time t, respectively. The DM test statistic is defined as:

$$\text{DM} = \sqrt{N}\frac{\hat{\mu}}{\hat{\sigma}} \sim \mathcal{N}(0,1), \tag{15}$$

where $\hat{\mu}$ is the sample mean and $\hat{\sigma}$ are the sample mean and sample standard deviation of the loss differential series, respectively, and N is the number of observations in the loss differential series.

The DM test is a two-tailed test, meaning that we are testing for the possibility that either model has better predictive accuracy. The null hypothesis is that both models have the same predictive accuracy, while the alternative hypothesis is that one model has better predictive accuracy than the other. If the p-value of the DM test is less than the significance level, $\alpha = 0.05$, we reject the null hypothesis and conclude that one model has better predictive accuracy than the other. On the other hand, if the p-value is greater than α, we fail to reject the null hypothesis and conclude that the two models have the same predictive accuracy.

Figs. 4 and 5 show the DM test statistics for the Central North Island and Lower South Island regions, respectively. We select these regions to illustrate the DM test results because the LE-GARCH-t is the best-performing model in the CNI, whereas the LEAR model is the best-performing model in the LSI region. We observe the test statistics rather than p-values because they provide a better visualization of the relative performance of the models. A negative test statistic lower than -1.67 indicates the row-wise model is statistically better, and a positive test statistics higher than 1.67 indicates the column-wise model is statistically better. In the heatmap, blue cells represent a better row-wise model, and red cells indicate a better column-wise model. Lighter colors indicate stronger significance in predictive accuracy (higher absolute test statistic). Gray-shaded cells indicate that there is no statistical significance between the predictive accuracy of the two models. A darker shade of gray corresponds to a lower test statistic, indicating stronger insignificance in predictive accuracy. The models in the heatmap are also sorted by their performance, from worst to best, in the corresponding region.

Sorting the models from worst to best based on their overall DM test scores provides a nice visual representation of the comparison between models. We notice the GARCH and GARCH-t models with all features included are generally outperformed by other models. On the other hand, the LE-GARCH-t and LE-SV-t are in the top three models for both regions. There are several other GARCH and SV models which are in the top ten in both regions.

The top performing model for CNI prices is the LE-GARCH-t model. However, according to the DM test, its predictive accuracy is statistically insignificant from the predictive accuracy of the LE-SV, LE-SV-t, and LEAR models. For LSI prices, the top performing model is the LEAR. However, the DM test suggests that its predictive accuracy is statistically insignificant from the predictive accuracy of the LE-SV-t, LE-GARCH-t, LE-SV, LE-GARCH, MI-SV-t, and the MI-SV models.

According to the DM test, we cannot conclude whether the GARCH-related or SV-related model is significantly better than the LEAR benchmark model. However, after observing the forecast metrics, we can confirm that these statistical models are on par with the LEAR, if not slightly better on the merit of forecast metrics. Additionally, the results show overwhelming evidence that the GARCH and SV models, and their t variants, provide significantly better forecasts when feature selection is conducted, as opposed to when all features are included.

5. Conclusion

This study presents a comparison of statistical models and machine learning models for daily electricity price forecasting in the New Zealand electricity market. We predict prices for the five encompassing regions in New Zealand. In particular, we compare the GARCH and SV models and their t-distribution variants with a variety of popular models in electricity price forecasting research, including LSTM, GRU, XGBoost, LEAR, and a four-layer DNN. The latter two models are considered benchmarks in this study. We use several exogenous variables for all models, including demand, generation fuel, forward prices and reserve prices, the HVDC transfer rate, and several weather-related variables. We also implement feature selection techniques, including LASSO, mutual information, and recursive feature elimination to create parsimonious feature sets for each region and each cross-validation set.

Our results suggest that the GARCH and SV statistical models, and their t-distributed variants, are very capable forecasting tools for EPF when paired with a variety of features, and feature selection methods to create an appropriate set of features. We find the LE-GARCH-t, the LE-SV-t, the LE-GARCH, the LE-SV to be consistently top-performing models, out-performing both the LEAR and DNN benchmark models, in terms of the forecast metrics considered. We also find their performance to increase by over 40% compared to the GARCH and SV models with all features included. This result speaks to the importance of implementing feature selection techniques.

We also investigated the Diebold–Mariano (DM) test results for all models considered. We found the difference in predictive accuracy of the top-performing GARCH and SV models to be statistically insignificant from the predictive accuracy of the LEAR model. This result suggests that the GARCH and SV models are not significantly better than the LEAR model. However, taking into account the forecast metrics, we can conclude that the GARCH and SV models are capable of out-performing the DNN, LSTM, GRU, and XGBoost models, in their simple forms. They are also capable of out-performing the LEAR model with respect to traditional forecast metrics.

A natural extension of our work would be to study more advanced statistical and machine-learning models. This includes the plethora of GARCH variants, such as the exponential GARCH (EGARCH), and the wide variety of novel machine-learning models that have been tested in recent literature. This could also include different classes of statistical models such as regime-switching and functional data models, see [51,52]. Another extension would be to study the performance of the models considered in this study on a more granular level, such as the hourly level. This would allow for a more detailed analysis of the performance of the models, and would also allow for a more detailed analysis of the performance of the models on different days of the week, and at different times of the day. Finally, an interesting study would involve the ensemble of these models to create hybrid models that are potentially capable of out-performing the individual models. A hybrid of statistical and machine learning models would also be an interesting study.

CRediT authorship contribution statement

Gaurav Kapoor: Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Writing – original draft. **Nuttanan Wichitaksorn:** Funding acquisition, Project administration, Resources, Supervision, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Gaurav Kapoor reports financial support was provided by Callaghan Innovation.

Data availability

Data will be made available on request

Acknowledgments

We thank the Editor-in-chief and the reviewers for their feedback that greatly improves the manuscript. This project was supported by the R&D Fellowship Grant from Callaghan Innovation (Grant Number GENED1801).

References

[1] Ministry of Business, Innovation and Employment. Energy in New Zealand 2020. 2021, [Online]. Available: https://www.mbie.govt.nz/dmsdocument/11679-energy-in-new-zealand-2020.
[2] Weron R. Electricity price forecasting: A review of the state-of-the-art with a look into the future. Int J Forecast 2014;30(4):1030–81.
[3] Hong T, Pinson P, Wang Y, Weron R, Yang D, Zareipour H. Energy forecasting: A review and outlook. IEEE Open Access J Power Energy 2020;7:376–88.
[4] Lago J, Marcjasz G, De Schutter B, Weron R. Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. Appl Energy 2021;293:116983.
[5] Diebold FX, Mariano RS. Comparing predictive accuracy. J Bus Econom Statist 2002;20(1):134–44.
[6] Bollerslev T. Generalized autoregressive conditional heteroskedasticity. J Econometrics 1986;31(3):307–27.
[7] Tan Z, Zhang J, Wang J, Xu J. Day-ahead electricity price forecasting using wavelet transform combined with ARIMA and GARCH models. Appl Energy 2010;87(11):3606–10.
[8] Knittel CR, Roberts MR. An empirical examination of restructured electricity prices. Energy Econ 2005;27(5):791–817.
[9] Diongue AK, Guegan D, Vignal B. Forecasting electricity spot market prices with a k-factor GIGARCH process. Appl Energy 2009;86(4):505–10.
[10] Wu L, Shahidehpour M. A hybrid model for day-ahead price forecasting. IEEE Trans Power Syst 2010;25(3):1519–30.
[11] Lago J, De Ridder F, De Schutter B. Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms. Appl Energy 2018;221:386–405.
[12] Gianfreda A, Grossi L. Forecasting Italian electricity zonal prices with exogenous variables. Energy Econ 2012;34(6):2228–39.
[13] Huurman C, Ravazzolo F, Zhou C. The power of weather. Comput Statist Data Anal 2012;56(11):3793–807.
[14] Karakatsani NV, Bunn DW. Fundamental and behavioural drivers of electricity price volatility. Stud Nonlinear Dyn Econom 2010;14(4).
[15] Sheppard K, Khrapov S, Lipták G, mikedeltalima, Capellini R, alejandro cermeno, Hugle, esvhd, bot S, Fortin A, JPN, Judell M, Li W, Adams A, jbrockmendel, Rabba M, Rose ME, Tretyak N, Rochette T, Leo U, RENE-CORAIL X, Du X, elik B. Bashtage/arch: Release 5.3.1. 2022, http://dx.doi.org/10.5281/zenodo.6684078, [Online]. Available:.
[16] Taylor SJ. Modeling stochastic volatility: A review and comparative study. Math Finance 1994;4(2):183–204.
[17] Ghysels E, Harvey AC, Renault E. Stochastic volatility. Handbook of Statist 1996;14:119–91.
[18] Kim S, Shephard N, Chib S. Stochastic volatility: likelihood inference and comparison with ARCH models. Rev Econom Stud 1998;65(3):361–93.
[19] Kastner G, Frühwirth-Schnatter S. Ancillarity-sufficiency interweaving strategy (ASIS) for boosting MCMC estimation of stochastic volatility models. Comput Statist Data Anal 2014;76:408–23.
[20] Chan JC. Moving average stochastic volatility models with application to inflation forecast. J Econometrics 2013;176(2):162–72.
[21] Chan JC, Grant AL. Modeling energy price dynamics: GARCH versus stochastic volatility. Energy Econ 2016;54:182–9.
[22] Tiwari AK, Kumar S, Pathak R. Modelling the dynamics of Bitcoin and Litecoin: GARCH versus stochastic volatility models. Appl Econ 2019;51(37):4073–82.
[23] Kastner G. Dealing with stochastic volatility in time series using the R package stochvol. 2019, arXiv preprint arXiv:1906.12134.
[24] Tibshirani R. Regression shrinkage and selection via the lasso. J R Stat Soc Ser B Stat Methodol 1996;58(1):267–88.
[25] Uniejewski B, Nowotarski J, Weron R. Automated variable selection and shrinkage for day-ahead electricity price forecasting. Energies 2016;9(8):621.
[26] Ziel F, Weron R. Day-ahead electricity price forecasting with high-dimensional structures: Univariate vs. multivariate modeling frameworks. Energy Econ 2018;70:396–420.
[27] Uniejewski B, Weron R. Efficient forecasting of electricity spot prices with expert and LASSO models. Energies 2018;11(8):2039.
[28] Luo S, Weng Y. A two-stage supervised learning approach for electricity price forecasting by leveraging different data sources. Appl Energy 2019;242:1497–512.
[29] Atef S, Eltawil AB. A comparative study using deep learning and support vector regression for electricity price forecasting in smart grids. In: 2019 IEEE 6th international conference on industrial engineering and applications (ICIEA). IEEE; 2019, p. 603–7.
[30] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–80.
[31] Cho K, Van Merriënboer B, Bahdanau D, Bengio Y. On the properties of neural machine translation: Encoder-decoder approaches. 2014, arXiv preprint arXiv:1409.1259.
[32] Coelho IM, Coelho VN, Luz EJdS, Ochi LS, Guimarães FG, Rios E. A GPU deep learning metaheuristic based model for time series forecasting. Appl Energy 2017;201:412–8.
[33] Fan C, Xiao F, Zhao Y. A short-term building cooling load prediction method using deep learning algorithms. Appl Energy 2017;195:222–33.
[34] Wang H-z, Li G-q, Wang G-b, Peng J-c, Jiang H, Liu Y-t. Deep learning based ensemble approach for probabilistic wind power forecasting. Appl Energy 2017;188:56–70.
[35] Chang Z, Zhang Y, Chen W. Electricity price prediction based on hybrid model of adam optimized LSTM neural network and wavelet transform. Energy 2019;187:115804.
[36] Huang C-J, Shen Y, Chen Y-H, Chen H-C. A novel hybrid deep neural network model for short-term electricity price forecasting. Int J Energy Res 2021;45(2):2511–32.
[37] Chen T, Guestrin C. Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016, p. 785–94.
[38] Breiman L. Classification and regression trees. Routledge; 2017.
[39] Bergstra J, Bardenet R, Bengio Y, Kégl B. Algorithms for hyper-parameter optimization. Adv Neural Inf Process Syst 2011;24.
[40] Bergstra J, Komer B, Eliasmith C, Yamins D, Cox DD. Hyperopt: a python library for model selection and hyperparameter optimization. Comput Sci Discov 2015;8(1):014008.
[41] Kozachenko LF, Leonenko NN. Sample estimate of the entropy of a random vector. Probl Pereda Inf 1987;23(2):9–16.
[42] Huang M-L, Hung Y-H, Lee W, Li R-K, Jiang B-R. SVM-RFE based feature selection and Taguchi parameters optimization for multiclass SVM classifier. Sci World J 2014;2014.
[43] Gholipour Khajeh M, Maleki A, Rosen MA, Ahmadi MH. Electricity price forecasting using neural networks with an improved iterative training algorithm. Int J Ambient Energy 2018;39(2):147–58.
[44] Ebrahimian H, Barmayoon S, Mohammadi M, Ghadimi N. The price prediction for the energy market based on a new method. Econ Res-Ekonomska IstraŽIvanja 2018;31(1):313–37.
[45] Naz A, Javed MU, Javaid N, Saba T, Alhussein M, Aurangzeb K. Short-term electric load and price forecasting using enhanced extreme learning machine optimization in smart grids. Energies 2019;12(5):866.
[46] Shao Z, Yang S, Gao F, Zhou K, Lin P. A new electricity price prediction strategy using mutual information-based SVM-RFE classification. Renew Sustain Energy Rev 2017;70:330–41.
[47] Wang H, Lei Z, Zhang X, Zhou B, Peng J. A review of deep learning for renewable energy forecasting. Energy Convers Manage 2019;198:111799.
[48] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014, arXiv preprint arXiv:1412.6980.
[49] Hyndman RJ, Koehler AB. Another look at measures of forecast accuracy. Int J Forecast 2006;22(4):679–88.
[50] Hyndman RJ, Athanasopoulos G. Forecasting: Principles and practice. OTexts; 2018.
[51] Shah I, Akbar S, Saba T, Ali S, Rehman A. Short-term forecasting for the electricity spot prices with extreme values treatment. IEEE Access 2021;9:105451–62.
[52] Jan F, Shah I, Ali S. Short-term electricity prices forecasting using functional time series analysis. Energies 2022;15(9):3423.