Journal of Physics:
Conference Series

PAPER • OPEN ACCESS

Enhancing the prediction of electric load demand: a comparative analysis of ARIMA and ANN models for the case of a small touristic island

To cite this article: Claudio Galli *et al* 2024 *J. Phys.: Conf. Ser.* **2893** 012120

Enhancing the prediction of electric load demand: a comparative analysis of ARIMA and ANN models for the case of a small touristic island

Claudio Galli[1], Francesco Superchi[1] and Alessandro Bianchini[1*]

[1] Department of Industrial Engineering, University of Florence, Florence, Italy

[*]E-mail: alessandro.bianchini@unifi.it

Abstract. Accurate prediction of energy demand is necessary for efficient power system operation, particularly in energy systems with high renewable sources integration. This study compares different methods for predicting load demand using statistical regression. In particular, the goal is to provide insights on the differences between simpler AutoRegressive Integrated Moving Average (ARIMA) and more complex Artificial Neural Network (ANN) models. The algorithms are used to predict electric load demand in Tilos, a small Greek island with strong seasonal trends due to summer tourism. The case study is significant as Tilos' outdated electrical grid must adapt to an increasing share of renewable energy sources, making load forecasting increasingly important. The algorithms were developed in Python using open-source tools (such as StatsModels and TensorFlow). Hyperparameters' tuning, crucial for enhancing forecasting effectiveness, was performed using stochastic optimization with Differential Evolution to minimize RMSE. The optimal normalized RMSE was reported as 9.72% for ANN and 9.54% for ARIMA, showing the effectiveness of both methods, with a slight edge for the statistical model. This work provides critical information regarding load prediction methodologies, highlighting practical guidance for energy managers, policymakers, and researchers in power system planning and operation.

1. Introduction

In the framework of energy management optimization, the rapid growth of intermittent renewable sources has introduced a high degree of unpredictability for grid operators. Therefore, while the introduction of energy storage systems is necessary, it still needs research and development, mostly regarding advanced control techniques to fully exploit their potential and optimize charge and discharge cycles. In turn, advanced load forecasting methods are needed for such methods to be implemented effectively.

1.1 Load forecasting: State of the Art

Multiple load forecasting techniques have been widely used and applied with various degrees of success. Mathematical models, such as linear regression (LR [1]) and Support Vector Regression (SVR [2]), have often been used with successful results to operate short-term load demand predictions.

Between the various statistical regression models, Auto Regressive Integrated Moving Average (ARIMA [3]) algorithms perform with high levels of accuracy. A seasonal exogenous ARIMA (SARIMAX) model for short-term load forecasting has been developed in [4] obtaining accurate forecasts for the following day with errors between 1.32% and 2.80%. Similarly, ARIMA load prediction was used in [5] in the context of energy management in buildings, reporting a 5.1% MAPE (Mean Absolute Percentage Error) for a full-year prediction.

Recently, the popularity of machine-learning prediction techniques has quickly increased due to their effectiveness and easy implementation [6]. In [7], the authors developed an ANN load prediction model for short-term forecasting, obtaining a MAPE of 1.38% for the following single-day prediction. Similarly an ANN was used in [8] to predict day-ahead demand load using temperature as input value, reporting an average absolute error of 2.06% for the 24-hour prediction.

Moreover, a direct comparison has been done in [9], where ANN reported generally better accuracy than the developed ARIMA model: such a result is, however, case-dependent and relative to the application itself. Even in [10], the two methods are compared, showing again better accuracy for the ANN model for most of the predictions.

The effectiveness of both algorithms is dictated by their hyperparameters' tuning, often user-defined. For this reason, authors in [11] used a genetic algorithm to optimize the neural network architecture. In this work, such a process is proposed and applied also to the SARIMAX model. After the optimization of the hyperparameters of both algorithms, a direct comparison is then carried out to figure out the best case-specific approach.

1.2 Case study

This study considers the historical load of Tilos, a small Greek island in the Mediterranean Sea. Tilos represents a very critical case study for various reasons. Indeed, the island is characterized by a large-size renewable installation, comprising an 800-kW wind turbine, 160-kWp of solar panels, and 2.88-MWh of storage systems in Zebra technology. This system serves as a Hybrid Power System (HPS) for energy export to the Kos-Kalymnos grid, following precise scheduling provided the day before. A full description of the system can be found in [12]. It is therefore crucial to develop an accurate load forecasting algorithm to optimally manage renewable energy and storage systems.

This development must start from the available measured load data. Two years of Tilos electric load demand (2021 and 2022) were provided by Eunice Energy Group, owner and manager of the HPS. The provided dataset reports the electric load demand with a single-minute frequency (Figure 1(a)).

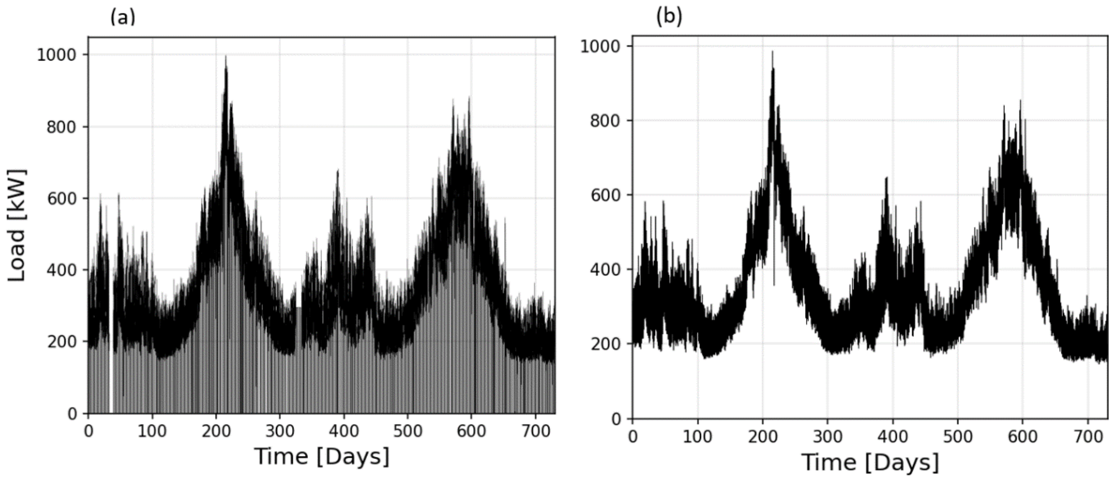

Figure 1. Two years of single-minute frequency electric load demand (original dataset, (a)) and cleaned hourly-averaged data (b).

The shape of the load presents an evident seasonal trend, with significantly higher demand during warmer months. This characteristic is due to tourism and provides a further degree of complexity that needs to be addressed by the forecasting algorithm. Moreover, measurement errors or interruptions are evident in the provided load demand dataset. Therefore, a further data cleaning process was carried out, whose details are reported in [13]. Since this work aims to forecast the average-hour demand load, the original dataset was then averaged. Figure 1(b) reports the final result after the cleaning and averaging process and does not present outliers or unusual behaviors in the trend of electric load demand.

2. Methodology

2.1 Load forecasting techniques
As mentioned in Section 1.1, the main techniques used for the load forecasting can be divided essentially into two categories: statistical regression models and machine learning algorithms [14]. This work focuses on the most widely applied algorithms: for statistical regression models, SARIMAX models are used; instead, for data-driven algorithms, fully connected artificial neural networks (FCNNs) are used. In particular, while for machine-learning models more advanced neural network structures are sometimes used to improve accuracy (such as LSTM, GRU, or recurrent neural networks, [15]), FCNNs were preferred among the various architectures for their effectiveness, simplicity, and lower computational cost compared to more advanced types [16].

Both algorithms are trained on the previous N days of operation to predict the following 24 hours i.e., the “predicted day”. Then, the training window shifts forward to test the algorithm on again N days before the second “predicted day”, and so on (see Figure 2). It is indeed key to train the algorithm on an operating window in proximity to the application itself. In this way, all the phenomena that are affecting the real operation are considered. On the other hand, training the algorithm with data far in time can induce bias errors in the prediction and capture trends unrelated to the current operation. For this reason, for both techniques the number of training days will be a variable to optimize in order to increase the prediction accuracy.

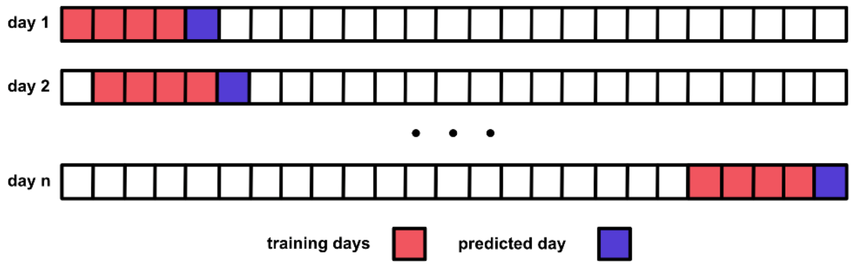

Figure 2. Moving window approach to test the actual performance of the algorithm.

The ARIMA algorithm is capable of predicting required parameters through a simple mathematical formula (Equations 1 and 2).

$$y_t^* = \Delta^d y_t \tag{1}$$

$$y_t^* = \mu + \sum_{i=1}^{p} \varphi_i y_{t-1}^* + \sum_{i=1}^{q} \theta_i \epsilon_{t-1} + \epsilon_t \tag{2}$$

The prediction is computed by introducing a white-noise function into a constant value (μ). The hyperparameters p, d, and q represent the order of the summation terms for AR (Auto regression), I (Integration), and MA (Moving average), respectively. These terms modify the model structure, and they are often user-defined:

- p modifies the number of the auto regression summation term.
- d defines the non-seasonal number of differentiations (used to achieve stationarity in the time series, needed for ARIMA application).
- q corresponds to the number of lagged forecast errors in the moving average term.

An ARIMA model can be improved with additional seasonality trends (SARIMA, which adds additional P, D, Q, and S orders) and by introducing exogenous values (ARIMAX) to generate a complete SARIMAX model. The detailed model description and working principles of SARIMAX can be found in [17].

Differently, an ANNs approach can perform a prediction after a training phase during which its parameters (i.e., how its neurons propagate the information) have been tuned to correlate the load with its future values. In this application, the FCNN hyperparameters tuned are the numbers of neurons and layers, the training days, and the training epochs. Similarly to the SARIMAX model, a detailed explanation of neural networks' working principles is outside the scope of this work and can be found in [18].

Lastly, for increased model accuracy, exogenous variables are introduced into the forecasting algorithms as inputs. After computing the Pearson correlation coefficient (PCC [19]), useful in highlighting correlations between certain variables and the load demand, six additional variables are introduced for prediction evaluation:

- Hour/Day: the hour of the day, in a 24-hour cycle.
- Day/Week: the day of the week, in a 7-day cycle.
- Weekend: a binary value, equal to 1 for Saturday and Sunday and 0 for other days.
- Lag1d: the hourly average load demand 24 hours before.
- Lag2d: the hourly average load demand 48 hours before.

- Lag7d: the hourly average load demand 168 hours (one week) before.

Table 1 reports the PCC between the considered time series. A very strong autocorrelation between the load and its lagged values is present, especially with the 24-hour lagged series (Lag1d). Moreover, while Day/Week and Weekend values do not show a strong correlation factor, their introduction provided improved accuracy in preliminary evaluation. Interestingly, the weekly correlation factor suggests an increase in load during weekends, highlighting once again the tourism importance and effect on the load profile.

Table 1. PCC values for the exogenous variable introduced in the prediction model.

Prediction model	Hour/Day	Day/Week	Weekend	Lag1d	Lag2d	Lag7d
PCC	0.139	0.006	0.005	0.971	0.952	0.898

2.2 Error metrics

To evaluate the accuracy of the prediction models, this work uses the following error metrics:

- *Normalized Root Mean Square Error (NRMSE)*: the main parameter for prediction error evaluation. Indeed, RMSE is the target of the minimization process carried out in the optimization process. As reported in [20], RMSE is often regarded as the most important error evaluation parameter for forecasting because of its sensitivity to bigger errors or outliers. NRMSE computation is reported in Equation 3:

$$NRMSE = \frac{1}{y_{AVG}} \cdot \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - p_i)^2} \tag{3}$$

  Where:
  - n is the number of predicted values, in this case the number of hours in a year (8760)
  - y_i is the real load demand value for every hour [kW]
  - y_{AVG} is the average value of the real load demand [kW]
  - p_i is the predicted load demand for every hour [kW]

- *Normalized Mean Absolute Error (NMAE)*: calculated as in Equation 4, this metric reports the average differences between each pair of predicted and actual values. Therefore, compared to RMSE, each error is treated independently from its size. This value therefore provides a clear indicator of the general error observed.

$$NMAE = \frac{1}{y_{AVG} \cdot n}\sum_{i=1}^{n}|y_i - p_i| \tag{4}$$

- *Forecasting bias (Bias)*: the forecasting bias, computed as in Equation 5, represents the mean of the prediction error. Maintaining the sign, it provides information about the trend to over or underestimate. This parameter is useful for implementing corrections.

$$bias = \frac{1}{n}\sum_{i=1}^{n}(y_i - p_i) \tag{5}$$

- *Maximum overshoot (err_{MAX+}) and maximum undershoot (err_{MAX-})*: the maximum error produced, both positive (Equation 6) and negative (Equation 7), by the prediction model. This value results of crucial importance. Depending on the application, a model with higher RMSE or MAE could be preferred over a more accurate one if maximum errors are lower.

$$err_{MAX+} = MAX(y_i - p_i) \tag{6}$$

$$err_{MAX-} = MIN(y_i - p_i) \tag{7}$$

2.3 Genetic algorithm for hyperparameters optimization

After providing a general overview of the two forecasting techniques, the models were integrated into an optimizer to obtain the best hyperparameters.

Due to the complexity of the optimization function, this work implements the Differential Evolution algorithm, which is an evolutionary algorithm developed by Storn and Price in 1995 [21] capable of finding the global minimum in a very complex optimization function and easily implementable in Python through SciPy libraries [22]. The optimization target is the minimization of the Root Mean Square Error (RMSE), a common accuracy estimation metric [20].

The hyperparameters to be optimized, their range and their resolution in the optimization process are reported in Table 2 for the SARIMAX model and Table 3 for the FCNN model.

Table 2. SARIMAX optimization parameters.

SARIMAX optimization	Order (p, d, q)	S order (P, D, Q, S)	Training days
MIN	(0, 1, 0)	(0, 0, 0, 0)	10
MAX	(10, 5, 10)	(0, 1, 0, 168)	200
Resolution	(1, 1, 1)	(0, 1, 0, 24)	10

Table 3. FCNN optimization parameters.

FCNN optimization	Neurons	Layers	Epochs	Training days
MIN	10	1	50	10
MAX	100	3	500	200
Resolution	10	1	50	10

In detail, the S order (Seasonal order) of the SARIMAX was set to deterministically vary from:

- (0, 0, 0, 0) to represent a non-seasonal model.
- (0, 1, 0, 24) to represent a daily-seasonal model.
- (0, 1, 0, 168) to represent a weekly-seasonal model.

This set of values was chosen because of the shape of the load demand and the limitation imposed by the high computational cost of the introduced seasonality. As described in Section 2.1, daily and weekly lagged load time series are included in the exogenous variables (Table 1) due to their strong PCC with the current load demand. To strengthen the seasonal link between current and past values, the S order that the SARIMAX can assume is again related to the daily and weekly seasonality.

For the FCNN hyperparameters instead, an additional degree of freedom is introduced in the architecture building algorithm: as a matter of fact, the optimizer can freely relocate the maximum number of neurons, as the 100 neurons could be for example split into three layers or put entirely into the first layer.

3. Results

After the hyperparameters optimization process, Table 4 reports the optimized configuration of the SARIMAX and FCNN model, providing the lowest RMSE of the predicted year (2022).

Table 4. SARIMAX and FCNN optimized configuration (lowest RMSE).

Prediction model	Order (p, d, q)	S. order (P, D, Q, S)	Neurons layout	Epochs	Training days
SARIMAX	(3, 1, 9)	(0, 1, 0, 24)	-	-	190
FCNN	-	-	(80, 20, 0)	100	180

Both models necessitate a high amount of training days, crucial for predicting data with strong yearly trends. The SARIMAX model also prefers to integrate a single-day seasonality: this is due to the combination of high daily trend in the load profile (as shown in Table 1) and the fact that past-week load is already integrated in the model via the exogenous variable Lag7d (Section 2.1). Differently, the ANN optimized configuration instead positions 80% of its available neurons in its first hidden layer: in this way, the network can better capture a broad range of features from the input data. This is particularly useful in the case of high-dimensional input data, which matches our setup considering the 180 training days and 6 exogenous variables, consisting of a total of 25920 observations for every predicted value.

With the optimized configuration, Table 5 reports error parameters resulting from the application of the forecast methods in the predicted year (2022) with day-ahead forecasting of one-hour average electric load demand.

Table 5. SARIMAX and FCNN estimation accuracy metrics and errors.

Prediction model	NRMSE	NMAE	bias [kW]	err_{MAX+} [kW]	err_{MAX-} [kW]
SARIMAX	**0.0955**	**0.0674**	**-0.090**	287	-247
FCNN	0.0972	0.0694	-1.211	**220**	**-222**

The optimized SARIMAX model presents a slightly lower RMSE (target error evaluation parameter), proving to be marginally better than the optimized FCNN. Even if both models report a general underestimation, the NMAE and the forecast bias results from the SARIMAX application are marginally lower compared to the FCNN prediction. On the other hand, results from SARIMAX show the highest under- and over-estimation error compared to the FCNN, which can be of critical importance depending on the model application. For the specific case considered in this work, RMSE and MAE are the most relevant accuracy metrics. Figure 3 compares the daily trend of actual and predicted values of the electrical load during the year 2022.

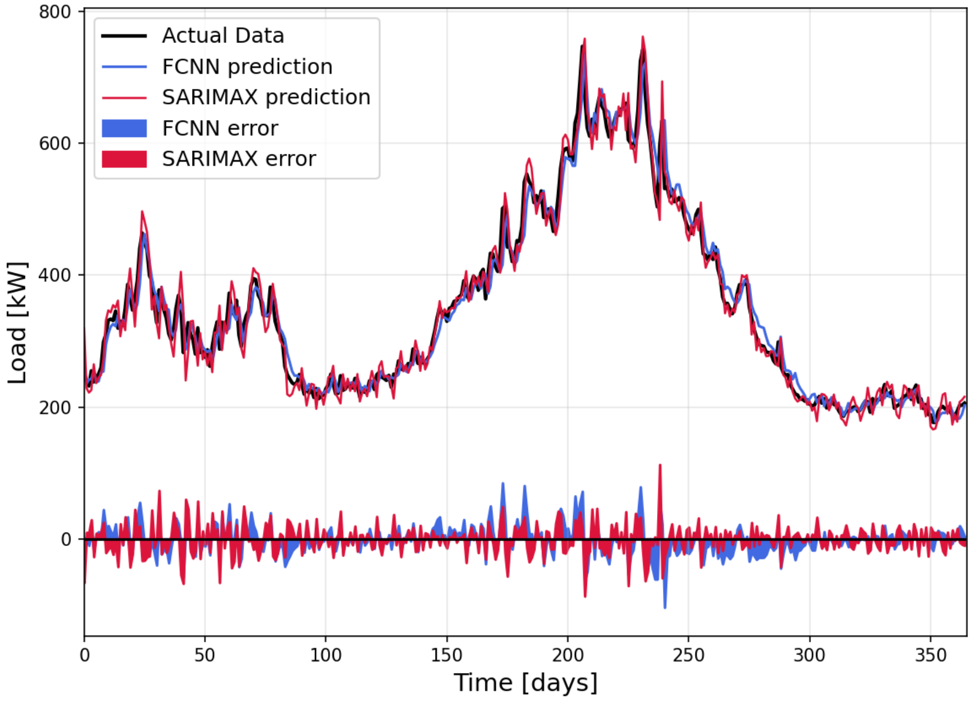

Figure 3. 24-hour moving average prediction and models' errors for the predicted year (2022).

Figure 4 shows the trend for two 14-day extracts of hourly predictions, where the forecasting process is noticeable. The figures show how the predicted load demand is often similar to the load of the previous day: this is due to the high correlation between the two. Unexpected peaks are the most trivial behavior to predict: when sudden changes from current day and past day data are present, the highest errors are reported (Day 107-108 in Figure 4(a) and Day 207-208 in Figure 4(b)). In particular, on Day 107 the daily load curve abruptly changes shape introducing an unexpected high peak of consumption during late morning. Because of the high correlation of the load trend between two subsequent days, the forecasting algorithm tries to mimic the same behavior on the day after the unexpected peak. Therefore, this event produces an additional error when the load shape returns to its normal trend. On the other hand, for significant but steadier changes, such as the growing trend from day 202 to day 207 (Figure 4(b)), both algorithms can follow the load, with the SARIMAX prediction model showing a more rapid and accurate response.

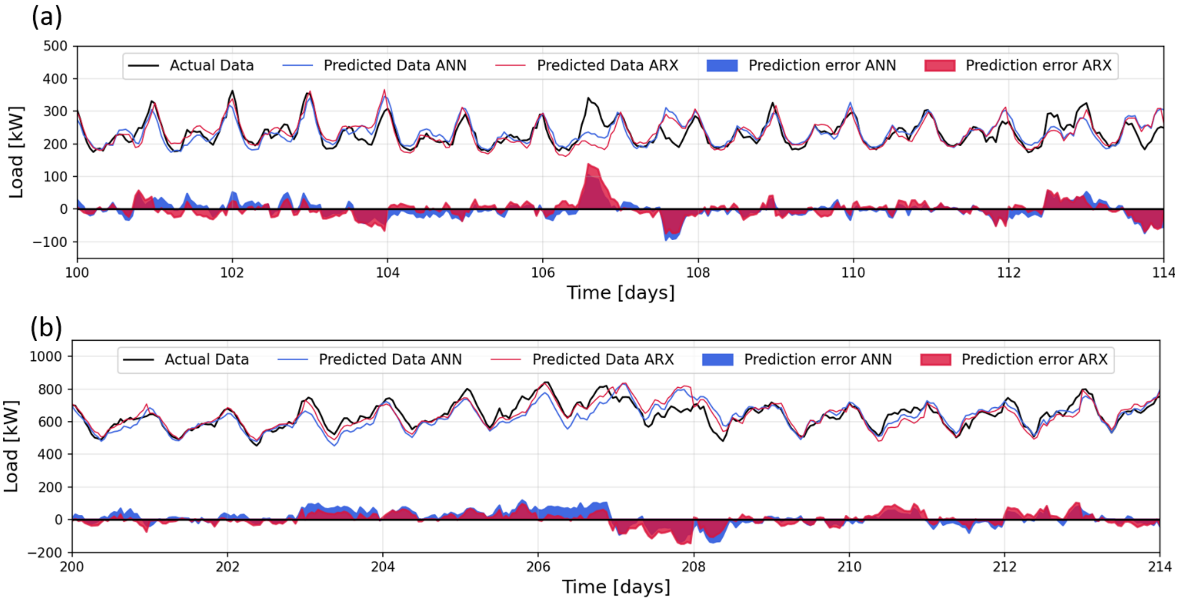

Figure 4. Average hourly load forecasting partial zoom of two different 14-day extracts of predicted year.

Histograms in Figure 5 show the statistical distribution of prediction errors, both on a linear and logarithmic scale of the y-axis (Figure 5(a) and Figure 5(b) respectively). Particularly, the histogram on the logarithmic scale (Figure 5(b)) clearly highlights the tails, showing the maximum errors that occurred using both prediction techniques.

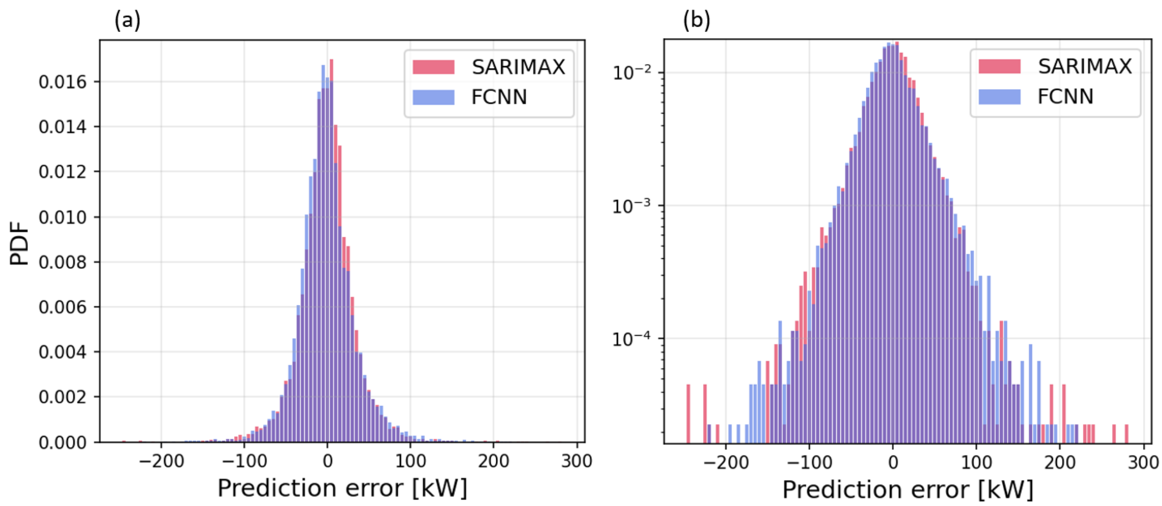

Figure 5. Histogram of the prediction error of SARIMAX and FCNN model, in linear (a) and log (b) y-scale to highlight histogram tails.

Interestingly, a specular behavior of the prediction error can be observed in the histograms, with over and underestimation errors being more prevalent either for the SARIMAX or the FCNN model depending on their intensity for low to medium values. Differently, for both positive and negative signs, high errors are more common in FCNN forecasting, but the maximum reported errors are in the SARIMAX prediction, as shown by the histogram tails in Figure 5(b).

4. Conclusions

This work assesses the criticalities and necessity of load forecasting algorithms by comparing two of the most common methods: SARIMAX (statistical regression) and FCNN (machine-learning). Those methods were applied to predict one year of day-ahead hourly averaged user load demand of Tilos, a Greek island in the Mediterranean Sea. Both algorithms are refitted or retrained daily, updating their coefficients and weights to reduce prediction error.

As both SARIMAX and FCNN models require user-defined hyperparameters that influence the model accuracy, an evolutionary algorithm (Differential Evolution) was integrated to optimize such hyperparameters in a global optimization process, minimizing the RMSE of the predicted year. Once optimized, both models reported a similar NRMSE: 9.54% for the SARIMAX model and 9.72% for the FCNN model. Some differences are highlighted, showing different prediction accuracies depending on the load demand. The statistical model reports the highest errors, while the data-driven approach proposes safer forecasting, missing some extreme values at the cost of a loss in accuracy.

Even if recent load-prediction state-of-the-art approaches have leaned mostly towards machine learning techniques, conventional statistical regression models may perform better for specific applications, if properly tuned. While satisfying levels of accuracy are obtained, further studies are required to increase prediction effectiveness by integrating different exogenous variables and using more advanced machine-learning approaches.

Lastly, while this work focuses on the most standard type of ANNs, advanced types (such as LSTM, GRU, and recurrent neural networks) will be considered in future studies, possibly providing additional improvements in accuracy.

Acknowledgments

Thanks are due to Eunice Energy Group, and particularly to Dr. George Pechlivanoglou, for providing access to the data of the Tilos Island in the framework of the joint research program connected to the PhD of Dr. Superchi.

References

[1] M. Mahmud, 'Isolated area load forecasting using linear regression analysis: practical approach', Sep. 2011, doi: 10.4236/epe.2011.34067'].
[2] L. W. Chong, D. Rengasamy, Y. W. Wong, and R. K. Rajkumar, 'Load prediction using support vector regression', in *TENCON 2017 - 2017 IEEE Region 10 Conference*, Nov. 2017, pp. 1069–1074. doi: 10.1109/TENCON.2017.8228016.
[3] R. H. Shumway and D. S. Stoffer, 'ARIMA Models', in *Time Series Analysis and Its Applications: With R Examples*, R. H. Shumway and D. S. Stoffer, Eds., Cham: Springer International Publishing, 2017, pp. 75–163. doi: 10.1007/978-3-319-52452-8_3.
[4] A. Tarsitano and I. L. Amerise, 'Short-term load forecasting using a two-stage sarimax model', *Energy*, vol. 133, pp. 108–114, Aug. 2017, doi: 10.1016/j.energy.2017.05.126.
[5] 'Electricity load forecasting using clustering and ARIMA model for energy management in buildings - Nepal - 2020 - JAPAN ARCHITECTURAL REVIEW - Wiley Online Library'. Accessed: Jun. 03, 2024. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1002/2475-8876.12135
[6] A. Baliyan, K. Gaurav, and S. K. Mishra, 'A Review of Short Term Load Forecasting using Artificial Neural Network Models', *Procedia Comput. Sci.*, vol. 48, pp. 121–125, Jan. 2015, doi: 10.1016/j.procs.2015.04.160.
[7] S. Singh, S. Hussain, and M. A. Bazaz, 'Short term load forecasting using artificial neural network', in *2017 Fourth International Conference on Image Information Processing (ICIIP)*, Dec. 2017, pp. 1–5. doi: 10.1109/ICIIP.2017.8313703.
[8] D. C. Park, M. A. El-Sharkawi, R. J. Marks, L. E. Atlas, and M. J. Damborg, 'Electric load forecasting using an artificial neural network', *IEEE Trans. Power Syst.*, vol. 6, no. 2, pp. 442–449, May 1991, doi: 10.1109/59.76685.

[9] G. P. Zhang, 'Time series forecasting using a hybrid ARIMA and neural network model', *Neurocomputing*, vol. 50, pp. 159–175, Jan. 2003, doi: 10.1016/S0925-2312(01)00702-0.
[10] Ö. Ö. Bozkurt, G. Biricik, and Z. C. Tayşi, 'Artificial neural network and SARIMA based models for power load forecasting in Turkish electricity market', *PLOS ONE*, vol. 12, no. 4, p. e0175915, Apr. 2017, doi: 10.1371/journal.pone.0175915.
[11] S. Nikbakht, C. Anitescu, and T. Rabczuk, 'Optimizing the neural network hyperparameters utilizing genetic algorithm', *J. Zhejiang Univ.-Sci. A*, vol. 22, no. 6, pp. 407–426, Jun. 2021, doi: 10.1631/jzus.A2000384.
[12] F. Superchi, N. Giovannini, A. Moustakis, G. Pechlivanoglou, and A. Bianchini, 'Optimization of the power output scheduling of a renewables-based hybrid power station using MILP approach: The case of Tilos island', *Renew. Energy*, vol. 220, p. 119685, Jan. 2024, doi: 10.1016/j.renene.2023.119685.
[13] 'Towards the Introduction of Green Hydrogen in the Energy Mix of Mediterranean Islands Through the Integration of Wind and Solar Power: A Techno-Economic Case Study', proceedings.com. Accessed: Jun. 05, 2024. [Online]. Available: https://www.proceedings.com/069564-0301.html
[14] N. Suhermi, Suhartono, D. D. Prastyo, and B. Ali, 'Roll motion prediction using a hybrid deep learning and ARIMA model', *Procedia Comput. Sci.*, vol. 144, pp. 251–258, 2018, doi: 10.1016/j.procs.2018.10.526.
[15] M. Abumohsen, A. Y. Owda, and M. Owda, 'Electrical Load Forecasting Using LSTM, GRU, and RNN Algorithms', *Energies*, vol. 16, no. 5, Art. no. 5, Jan. 2023, doi: 10.3390/en16052283.
[16] A. Borovykh, C. W. Oosterlee, and S. M. Bohté, 'Generalization in fully-connected neural networks for time series forecasting', *J. Comput. Sci.*, vol. 36, p. 101020, Sep. 2019, doi: 10.1016/j.jocs.2019.07.007.
[17] S. Mulla, C. B. Pande, and S. K. Singh, 'Times Series Forecasting of Monthly Rainfall using Seasonal Auto Regressive Integrated Moving Average with EXogenous Variables (SARIMAX) Model', *Water Resour. Manag.*, vol. 38, no. 6, pp. 1825–1846, Apr. 2024, doi: 10.1007/s11269-024-03756-5.
[18] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, 'Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks', in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015, pp. 4580–4584. doi: 10.1109/ICASSP.2015.7178838.
[19] P. Schober, C. Boer, and L. A. Schwarte, 'Correlation Coefficients: Appropriate Use and Interpretation', *Anesth. Analg.*, vol. 126, no. 5, p. 1763, May 2018, doi: 10.1213/ANE.0000000000002864.
[20] M. Jain, S. Manandhar, Y. H. Lee, S. Winkler, and S. Dev, 'Forecasting Precipitable Water Vapor Using LSTMs', in *2020 IEEE USNC-CNC-URSI North American Radio Science Meeting (Joint with AP-S Symposium)*, Jul. 2020, pp. 147–148. doi: 10.23919/USNC/URSI49741.2020.9321614.
[21] S. Das and P. N. Suganthan, 'Differential Evolution: A Survey of the State-of-the-Art', *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011, doi: 10.1109/TEVC.2010.2059031.
[22] P. Virtanen *et al.*, 'SciPy 1.0: fundamental algorithms for scientific computing in Python', *Nat. Methods*, vol. 17, no. 3, pp. 261–272, Mar. 2020, doi: 10.1038/s41592-019-0686-2.