International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)
ISSN(P): 2249–6890; ISSN(E): 2249–8001
Vol. 10, Issue 3, Jun 2020, 12479–12488
© TJPRC Pvt. Ltd.

OriginalArticle

FORECASTING PHOTOVOLTAIC POWER USING BAGGING FEED-FORWARD NEURAL NETWORK

E. J. ZARATE[1], M. PALUMBO[2], A. L. T. MOTTA[3] & J. H GRADOS[4]

[1]*Department of Engineering, Universidad Privada Del Norte (UPN).*

[2]*Department Architecture, Urbanism and Construction, Universitat Politècnica de Catalunya (UPC).*

[3]*Postgraduate Program in Civil Engineering, Universidade Federal Fluminense (UFF).*

[4]*Department of Electrical and Electronic Engineering, Universidad Nacional del Callao (UNAC).*

ABSTRACT

This paper presents a forecast model of the active power of a photovoltaic (PV) power generation system. In this model, a feed-forward neural network (FNN) is combined with bootstrap aggregation techniques using the Box–Cox transformation, seasonal and trend decomposition using Loess, and a moving block bootstrap (MBB) technique. An analysis is conducted using the data provided by the active power of the PV power generation system; the data are collected every 30 min for 12 months. The FNN method combined with MBB techniques consistently outperformed the original FNN in terms of forecasting accuracy based on the root mean squared error, on the forecast from one day of anticipation. The results are statistically significant as demonstrated through the Ljung–Box test, which verifies that the forecast errors are not correlated, thereby validating the proposed model.

KEYWORDS: ***Bootstrap, Demand Forecasting, Energy Consumption, Feed-Forward Neural Networks, Photovoltaic Systems***

Received: Jun 08, 2020; **Accepted:** Jun 28, 2020; **Published:** Sep 12, 2020; **Paper Id.:** IJMPERDJUN20201188

INTRODUCTION

Electricity demand growth is particularly relevant in the residential sector, which currently uses approximately 40% of global energy resources and generates approximately one-third of the greenhouse gas emissions [1]. In addition to the risk of an energy crisis owing to the exclusive use of conventional energy generation, a marked variability in demand exists owing to variations in daily and seasonal environmental conditions, thereby creating simultaneity problems between demand and energy production [2]. Electricity microgeneration systems using photovoltaic (PV) panels are the most appropriate technical alternative for this problem [3]. The use of PV power plants and the ability to forecast the generation of solar energy can enhance the reliability of the system and reduce the price of energy, thereby improving electricity production planning [4]. Therefore, forecasting the production of PV energy is crucial for improving the integration of this sustainable energy.

In this regard, energy production is forecasted in terms of minutes, hours, days, months, or years to ensure demand coverage with the highest possible productivity margin [5]. Furthermore, the selection of the most suitable forecasting method depends on different factors, such as the complexity, nature, and purpose of the installed system, or the precision and adaptability of the estimation based on forecast, among others [6].A systematic review that identified several forecasting methods was presented by Jimenez et al. (2019) [5]. The results obtained by the authors indicate a classification based on physical methods established in satellite image data, solar irradiance, and cloud index forecast. In the forecasting process, the precision of time series methods such as autoregressive moving average, autoregressive integrated moving average, and space–time autoregressive moving average models are restricted by the nonlinearity of cloud movement and climatic fluctuations.

In this regard, the author emphasizes that artificial neural networks (ANNs) and support vector machines are more reliable solutions for predicting the production of PV energy in terms of global and horizontal solar irradiance [5]. This is in accordance with other studies [7], [8], [9], [10], [11], [12], [13], in which ANN methods have been used extensively to forecast the demand and production of electricity, owing to their better performances.However, for a high-frequency time series (i.e., daily and hourly recordings, etc.), the variance changes over time and in a nonsystematic manner in many cases, such that the high and low variance periods alternate. This generates a higher level of particularity and, consequently, a higher level of time series noise. The noise level affects the ANN learning process, decreasing the generalization capacity and causing overfitting. However, the implicationsof this might be reduced by stabilizing and smoothing the variation in the time series using bootstrap techniques [14].

Some studies showed a high predictive power and fewer errors in the bootstrap technique when the latter was used with other forecasting methods, e.g., a random forest algorithm was used in [15]. Therefore, the objective of this study is to determine the forecasting performance in a hybrid form of the ANN feed-forward model and bootstrap techniques using the active power data of a PV system.

METHODOLOGY

Fig. 1 shows the steps performed to obtain the result analysis. First, a Box–Cox transformation was performed, followed by a seasonal-trend decomposition using Loess (STL) of the sample into three parts: seasonal, trend, and remainder. Next, 100 subseries of the residual part were created using the bootstrap technique. Subsequently, each one of them was joined with the original seasonal and trend components, in which a reverse Box–Cox transformation was applied to them to obtain 100 versions of bootstrap series identical to the original. Next, forecasts were made with each of the bootstrap series using the feed-forward ANN model. Finally, the average of the 100 forecasts was obtained for the final forecast.

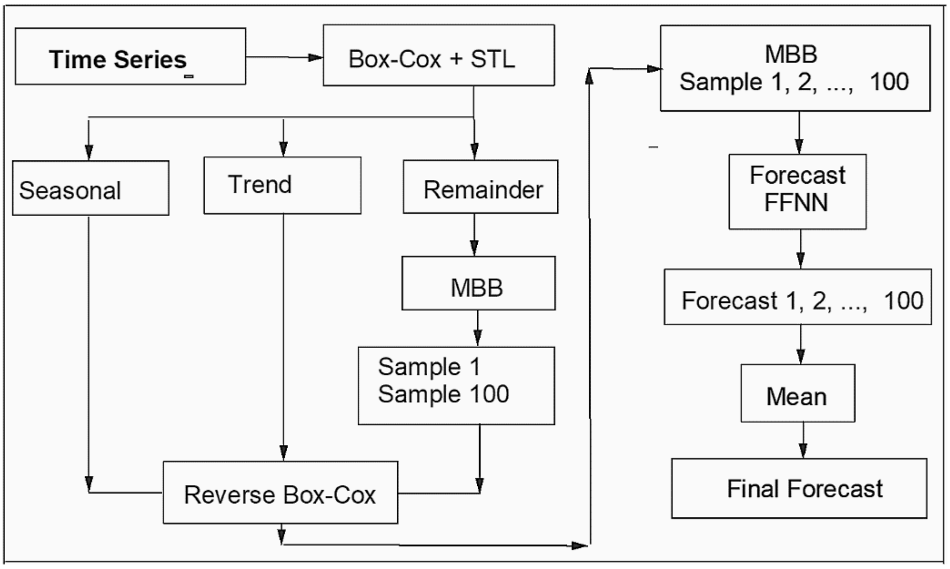

Figure 1: Illustration of methodology sequence.

A.Data Collection

Fig. 2 shows the active power data of a 3.63 kWp PV system, in the state of Rio de Janeiro, Brazil. The data sample selected included data for an entire year, i.e., the series started on January 1, 2018, from 00:00 and ended on December 31 (23:59) of the same year. The data were collected with readings taken every 10 s and averages of the values were obtained every 30 min. The data collected were stored in the EnergyLOG plus measurement device,and were then transferred to a personal computer.

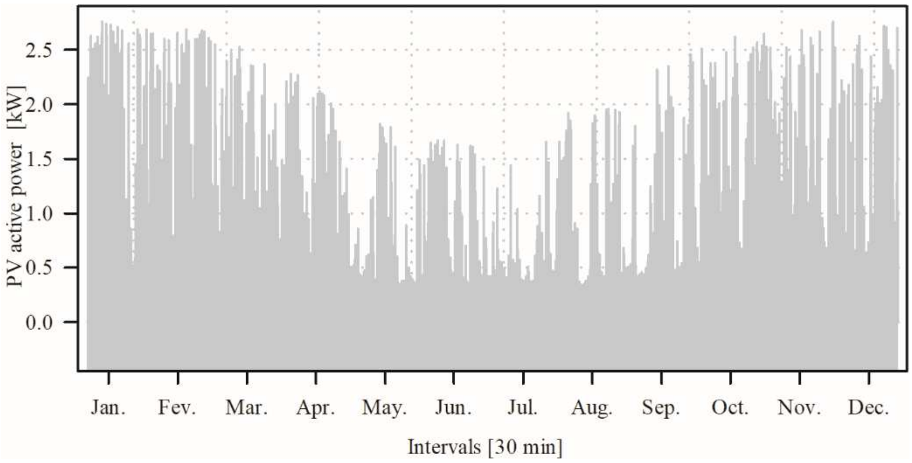

Figure 2: PV active power data used to evaluate the performance of the forecast model.

The 3.63 kWp PV solar system comprised 11 polycrystalline silicon modules. They formed groups in series and in parallel to satisfy the voltage and current requirements of the inverter. The main characteristics of the panel are shown below in Table I. Hence, the characteristics of the inverter are shown in Table II.

Table 1: Electrical Data Pv Module Under Stl

Type	POLY MODULE 330P		
	Specification	***Data***	***Unit***
1	Nominal Max. Power (Pmax)	330	W
2	Opt. Operating Voltage (Vmp)	33.9	V
3	Opt. Operating Current (Imp)	9.74	A
4	Open Circuit Voltage (Voc)	41.1	V
5	Module Efficiency	19.9	V

Table 2: Table Type Styles

Type	Growatt 3000TL		
	Specification	**Data**	**Unit**
Input Data	Max. DC power	3200	W
	Max. DC voltage	500	V
	Full load Mpp-Voltage range	250– 450	V
	PV voltage range MPPT	120–450	V
Output (AC)	Nominal AC output power	2850	W
	Max. AC power	3000	W
	Max. output current	13	A
	AC grid frequency range	50, 60 ±5%	Hz

Type	Growatt 3000TL		
	Specification	Data	Unit
Efficiency	Max.efficiency	97	%

B. Box–Cox Transformation

Box–Cox transformations include logarithms and power transformations, which are determined by the parameter λ [16]. These transformations are typically used to stabilize the variance of a time series and were originally proposed in [16]. Next, the Box–Cox transformation in (1) is presented.

$$\omega_t = \begin{cases} Log(y_t), & \lambda = 0; \\ \dfrac{\left(y_t{}^{\lambda} - 1\right)}{\lambda}, & \lambda \neq 0. \end{cases} \quad (1)$$

where y_t is the original data of the sample shown in Fig. 2, and ω_t is the transformed data. The Box–Cox transformation is the identity when λ = 1, it is the logarithm when λ = 0, or a transformation within the given range. The λ parameter is restricted to the range [0;1], and the Guerrero method was used to select its value, as described in [16].

C. STL Decomposition

Once the transformed data (ωt) were obtained, they showed various patterns. Hence, it was convenient to divide them into different components. In this study, STL was used because it is a powerful method for decomposing time series [17]. STL divides the time series into its trend, seasonal, and remainder components. The division is additive, as indicated in (2) [16]. The stl () in the R Team Core function was used to decompose the time series into their seasonal, trend, and remainder components [18].

$$\omega_t = S_t + T_t + R_t, \quad (2)$$

where ω_t is the time series data transformed; St, Tt, and Rt are the seasonal, trend-cycle, and remainder components, respectively.

D. Bootstrapped Remainder

A requirement for the forecast with time series is the stationarity of the data collected because they are typically correlated. It was achieved by initializing the remainder (R_t) using the moving block bootstrap (MBB) technique; the procedure described in [19] was used. For the active power PV series with a length n and a block size of L, [n / L] + 2 blocks from the remainder subseries of the STL decomposition were used. Subsequently, to obtain a series of the same length as the sample, the maximum possible values were discarded to reach the required length [19]. After performing the initialization of the residual (R_t), each bootstrap subseries was joined to the seasonal and trend components, yielding a total of 100 similar series. Finally, the inverse Box–Cox transformation was performed in each of the bootstrap series, and 100 similar series were obtained for the original (xi, t). In R, the bld.mbb.bootstrap ( ) function was used.

E. Artificial Neural Networks

1) Neural Network Architecture

Once the initialized series ($x_{i,\ t}$) was performed, forecast was performed for each one. In this study, a feed-forward neural network (FNN) model comprising different layers was used, where each layer of nodes received inputs from the previous layers, and these were input for the next layer. Next, the inputs for each node were combined via a weighted linear interaction [16]. The product was modified using a nonlinear function before being generated. The inputs to the hidden neuron, j, in Fig. 3, were combined linearly, as expressed in (3) [16].

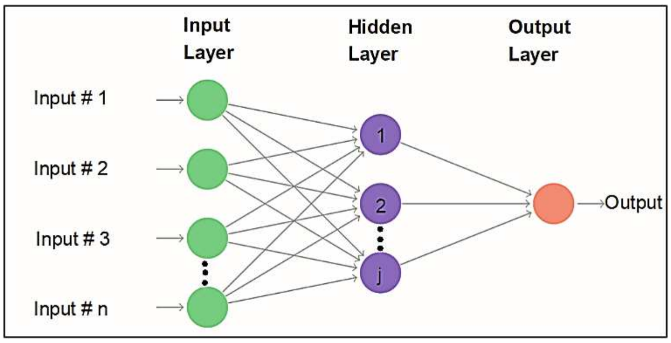

Figure 4: Feed-forward ANN with a hidden layer.

$$z_j = b_j + \sum_{i=1}^{n} w_{i,j} x_{i,t}, \quad (3)$$

where z_j corresponds to the level of internal activity of the neuron, which is obtained by combining the sum of the weighted inputs ($w_{i,\, j,}\ x_{i,\, t}$) with another adjustable value called bias (b_j). The weighted entries comprised synaptic weights ($w_{i,\, j}$) and input data ($x_{i,\, t}$). Therefore, parameters b_j and $w_{i,\, j}$ were obtained during the training phase, in which the network learned to use the observed data [16].

FNNs have been shown effective in solving complex tasks. The theoretical foundations for the solution of learning tasks are attributed to the universal approximation theorem. This theorem imposes a nonlinearity condition on hidden nodes, i.e., the function must be continuous, limited, increase uniformly [20].In this context, the derivative of the activation function is required by the weight update rules, and the differentiation condition becomes an algorithmic requirement. Therefore, the sigmoid function was used in the forecast model because it exhibits these properties [20]. The output (s (z)) of the linear combination is then modified using a nonlinear function before the exit of each layer using the sigmoid function, S (z), which is shown in (4) [16].

$$s_{(z)} = \frac{1}{1 + e^{-z_j}} \quad (4)$$

2) Neural Network Autoregression

The time series data can be used as input to an FNN, as shown in (5) [21]. In this study, an FNN model with a hidden layer was considered, and it is presented as FNN (p, P, k) [f]. This indicates that p non-seasonal lagged inputs, P seasonal lagged inputs, and k neurons exist in the hidden layer; additionally, [f] is expressed by the frequency.

$$\left(x_{i,t-1}, x_{i,t-2}, \dots, x_{i,t-p}, x_{i,t-f}, x_{i,t-2f}, x_{i,t-Pf}\right) \qquad (5)$$

The *nnetar ()* function in R [18] fits the FNN (p, P, k) f model. For this function, when the p and P values are not specified, they are selected automatically, i.e., for the seasonal time series, P = 1; furthermore, p is selected from the appropriate linear model according to the Akaike information criterion. If k is not specified, it is defined as (p + P + 1) / 2, approaching the nearest integer value. Finally, as the data presented in Fig. 2 were obtained with values read every 30 min, the daily frequency [f] was 48.

3) Learning Process of Neural Networks

The p, P, and k values in the learning process of the FNN were used for the network training. Therefore, the PV active power data were separated into training and test data. The size of the dataset for training was determined to be 80% [16]. Once the parameters that fit the model with the nnetar () function were obtained for both the original series (yt) and the initialized series (xi, t), they were verified by tabbing the number of internal neurons (k), recording the mean squared error (MSE) for each case. The MSE is the performance function used by the neural network.

4) Evaluation of Forecast Accuracy

Once the parameters of the FNN models were verified, both the original series data and the bootstrap series were used, and forecasts were made to determine the performance. Finally, the performance of the forecasts was evaluated using the procedure proposed, which involves performing a cross-validation of the forecasts [22].The two most used independent scale measures are based on absolute errors or square errors and are expressed by the mean absolute percentage error (MAPE) [23] and root mean square error (RMSE) [12] respectively, as shown in (6) and (7), respectively.

$$MAPE = \left(\sum_{t=T+1}^{T+h} \left| \frac{e_t}{y_t} \right| / h \right) x\ 100\% \quad (6)$$

$$RMSE = \sqrt{\sum_{t=T+1}^{T=h} (e_t)^2 / h,} \quad (7)$$

where the estimation of the forecast error (e_t) is expressed by the difference between the observed value (y_t) and its forecast (et), i.e., the unpredictable part of the observation, as indicated in (8). Furthermore, (h) is the number of cases evaluated [16].

$$e_t = \hat{y}_{t-}y_t \quad (8)$$

If the models do not fit the data and result in strongly correlated errors, this occurrence can be easily perceived by verifying the residuals for the serial correlation using the Ljung–Box test. The Ljung–Box test statistic has a chi-square distribution with u – n degrees of freedom and is expressed as shown in (9) [22].

$$Q = N(N+2) \textstyle\sum_{i=1}^{u} (N-u)^{-1} r_u^2, \quad (9)$$

where u is the number of lags used, i.e., 15; r_u is the estimated value of the u umpteenth serial autocorrelation coefficient; the p-value is expressed as Prob [Q ≥ χ2G.L]. The Ljung–Box test was implemented in R using the Box.test () function [18].

RESULTS AND DISCUSSIONS

In this section, the results of the forecast of active PV power are presented. For this purpose, the FNN (30, 1, 16) 48 and MBB-FNN (30, 1, 15) 48 models were obtained for the original data (y_t) and initialized series ($x_{i,\ t}$), respectively. The verification of these models based on the tabulation of the number of neurons in the hidden layer (k) is shown in Fig. 4. As shown, the number of hidden neurons for the FNN and MBB-FNN models were 16 and 15, respectively, as determined by the MSE registry.

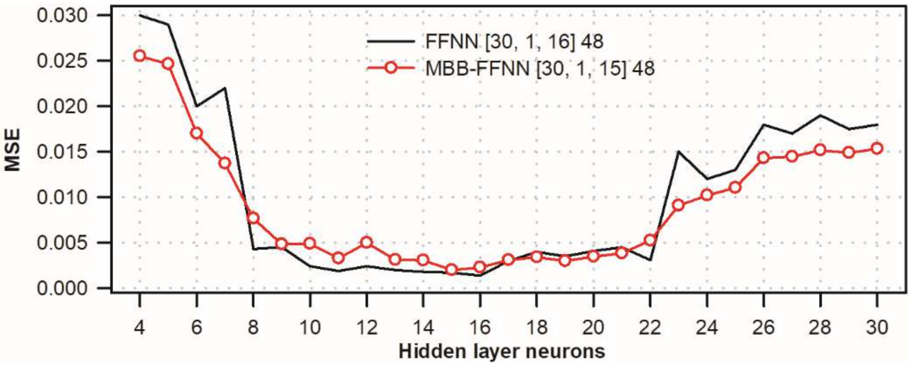

Figure 5: The number of hidden neurons for the FNN and MBB-FNN models.

In Fig. 5, the results of the forecasts are presented based on the performance of the defined forecast models, both for the original and initialized series. The results for three weeks of different months, randomly selected from the forecast set, are shown. The first week corresponds to the period from February 19 to 25, the second from June 2 to 8, and the last from December 23 to 29, 2018, for the FNN and MBB-FNN models.In Fig. 5, the forecast with the resampling data from the original series based on the MBB technique show a better generalization that is similar to the real data. For the February and June forecasts, the FNN model performed well although not better than the MBB-FNN model. However, the performance in December was worse for the FNN, whereas the MBB-FNN demonstrated better performance owing to its better generalization, based on the resampling of the time series.

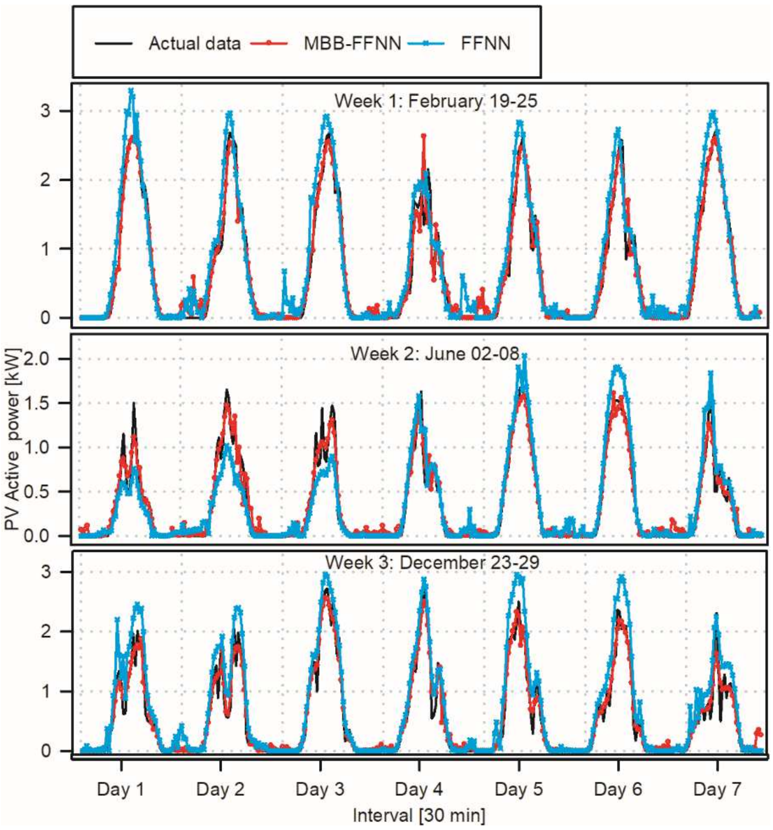

Figure 6: Forecasting from FNN and MBB-FNN models.

Consequently, Fig. 6 presents the results of the model performance. The MBB-FNN and FNN models demonstrated RMSE values of 12.15 × 10^{-2} and 11.13 × 10^{-2} W, respectively. This difference might be because the generated bootstrap series was previously smoothed, resulting in a slightly higher RMSE, as shown in the selection of the number of hidden neurons in Fig. 4. However, in the evaluation of the performance of the forecast, MBB-FNN demonstrated a better forecast consistency on the horizon than the FNN model, with RMSE values of 13.46 × 10^{-2} and 17.66 × 10^{-2} W, respectively.

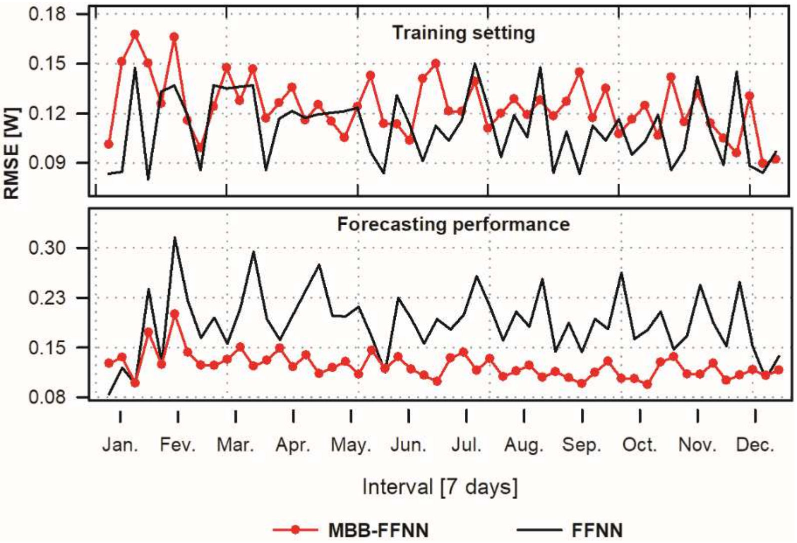

Figure 7: Forecasting performances of FNN and MBB-FNN models.

For the validation based on the MAPE indicator in percentage, the performances of the FNN and MBB-FNN models were 4.48% and 3.14%, respectively. Therefore, it can be affirmed that, when using the resampling of residuals in the time series using the MBB technique, uncertainties in the forecast active PV power reduced significantly. This is consistent with [15], where the author demonstrated high predictive power and less forecast uncertainty when using the MBB technique with the random forest algorithm method. Similarly, the result coincides with those of other studies using the bootstrap approach with the autoregressive integrated moving averagemodel, which was used for the monthly forecast of electrical loads [19]. The results show that combining these techniques with the FNN model for forecasting every 30 min substantially improved the performance.

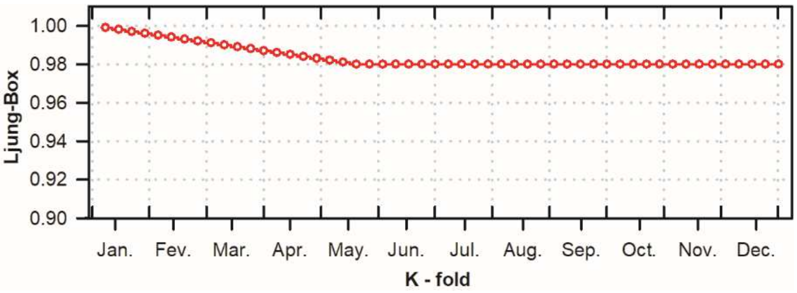

Figure 8: Ljung–Box test for the MBB-FNN model.

Finally, considering the better forecasting performance of the MBB-FNN model, the results obtained from the Ljung–Box test was then verified, as shown in Fig. 7. The results show that the mean of the values obtained for the entire horizon was close to unity, indicating that the results are statistically significant, thereby verifying the non-correlation of the forecast errors and validating the proposed model.

CONCLUSIONS

Herein, the modeling of the forecast of PV active power was presented. As demonstrated, the resampling of the time series using the bootstrap technique made it possible to obtain a better performance of the FNN. Furthermore, stability against forecast uncertainties was demonstrated because the MBB techniques guaranteed the generalization of forecasts when initializing the residual of the time series.

The FNN, combined with MBB techniques, consistently outperformed the forecast accuracy compared with the original FNN based on the RMSE and MAPE values, in predicting a day of anticipation with a frequency of 30 min. The results were statistically significant, as demonstrated through the Ljung–Box test, thereby verifying the non-correlation of the forecast errors and validating the proposed model.

ACKNOWLEDGEMENTS

The authors would like to thank the support by the Universidad Privada del Norte (UPN), Peru, and the Coordination for Higher Education Staff Development (CAPES - Brazil) for the assistance in this study.

REFERENCES

1. *M. L. Chalal, M. Benachir, M. White, and R. Shrahily, "Energy planning and forecasting approaches for supporting physical improvement strategies in the building sector: A review," Renew. Sustain. Energy Rev., vol. 64, pp. 761–776, Oct. 2016, doi: 10.1016/J.RSER.2016.06.040.*

2. *F. Yutaka Kuwabata Takigawa, R. Cavalcante Fernandes, E. Antonio Cardoso Aranha Neto, D. Tenfen, and E. Taghori Sica, "Energy Management by the Consumer with Photovoltaic Generation: Brazilian Market," IEEE Lat. Am. Trans., vol. 14, no. 5, pp. 2226–2232, May 2016, doi: 10.1109/TLA.2016.7530417.*

3. *T. Vieira da Silva, R. Vitor Arantes Monteiro, F. A. M. Moura, M. R. M. C. Albertini, M. A. Tamashiro, and G. Caixeta Guimaraes, "Performance Analysis of Neural Network Training Algorithms and Support Vector Machine for Power Generation Forecast of Photovoltaic Panel," IEEE Lat. Am. Trans., vol. 15, no. 6, pp. 1091–1100, Jun. 2017, doi: 10.1109/TLA.2017.7932697.*

4. *D. I. Jurj, D. D. Micu, and A. Muresan, "Overview of Electrical Energy Forecasting Methods and Models in Renewable Energy," in 2018 International Conference and Exposition on Electrical And Power Engineering (EPE), Oct. 2018, pp. 0087–0090, doi: 10.1109/ICEPE.2018.8559807.*

5. *J. Jimenez, A. Pertuz, C. Quintero, and J. Montana, "Multivariate Statistical Analysis based Methodology for Long-Term Demand Forecasting," IEEE Lat. Am. Trans., vol. 17, no. 01, pp. 93–101, Jan. 2019, doi: 10.1109/TLA.2019.8826700.*

6. *K. B. Debnath and M. Mourshed, "Forecasting methods in energy planning models," Renew. Sustain. Energy Rev., vol. 88, pp. 297–325, May 2018, doi: 10.1016/J.RSER.2018.02.002.*

7. *K. A. Kumari, N. Kumar Boiroju, T. Ganesh, and & P. Rajashekara Reddy, "FORECASTING SURFACE AIR TEMPERATURE USING NEURAL NETWORKS," Int. J. Math. Comput. Appl. Res., vol. 3, pp. 65–78, Jun. 2013.*

8. *L. Kondru and N. Rao, "DAMAGE DETECTION IN CANTILEVER BEAMS USING ARTIFICIAL NEURAL NETWORKS," Int. J. Mech. Prod. Eng. Res. Dev. , vol. 3–1, pp. 259–268, Mar. 2013.*

9. *V. Singh, "Application of Artificial Neural Networks for Predicting Generated Wind Power," Int. J. Adv. Comput. Sci. Appl., vol. 7, no. 3, 2016, doi: 10.14569/ijacsa.2016.070336.*

10. *N. M., A. Al, and S. A., "Load Balancing with Neural Network," Int. J. Adv. Comput. Sci. Appl., vol. 4, no. 10, 2013, doi:*

10.14569/ijacsa.2013.041021.

11. *A. H. Neto and F. A. S. Fiorelli, "Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption," Energy Build., vol. 40, no. 12, pp. 2169–2176, Jan. 2008, doi: 10.1016/J.ENBUILD.2008.06.013.*

12. *S. A. Kakar et al., "Artificial Neural Network based Weather Prediction using Back Propagation Technique," 2018. Accessed: Aug. 25, 2020. [Online]. Available: www.ijacsa.thesai.org.*

13. *A. F. Sheta, S. Elsir, and M. Ahmed, "A Comparison between Regression, Artificial Neural Networks and Support Vector Machines for Predicting Stock Market Index," 2015. Accessed: Aug. 25, 2020. [Online]. Available: www.ijarai.thesai.org.*

14. *D.-H. Le, "LINEAR REGRESSION AND ARTIFICIAL NEURAL NETWORKS FOR MODELING COMPRESSIVE STRENGTH OF SOIL-BASED CLSMS," Int. J. Civil, Struct. Environ. Infrastruct. Eng. Res. Dev. , vol. 5, pp. 25–34, Apr. 2015, Accessed: Sep. 04, 2020. [Online]. Available: www.tjprc.org.*

15. *Jian Zheng, Cencen Xu, Ziang Zhang, and Xiaohua Li, "Electric load forecasting in smart grids using Long-Short-Term-Memory based Recurrent Neural Network," in 2017 51st Annual Conference on Information Sciences and Systems (CISS), Mar. 2017, pp. 1–6, doi: 10.1109/CISS.2017.7926112.*

16. *H. Byeon, "Developing A Model for Predicting the Speech Intelligibility of South Korean Children with Cochlear Implantation using a Random Forest Algorithm," Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 11, pp. 88–93, 2018, doi: 10.14569/IJACSA.2018.091113.*

17. *R. J. Hyndman and G. Athanasopoulos, Forecasting : principles and practice, 2nd ed. OTexts, 2018.*

18. *R. Cleveland, W. Cleveland, J. McRae, and I. Terpenning, "STL: A seasonal-trend decomposition procedure based on Loess.," J. Off. Stat., vol. 6. No 1, pp. 3–73, Jan. 1990, Accessed: May 14, 2018. [Online]. Available: http://www.scb.se/contentassets/ca21efb41fee47d293bbee5bf7be7fb3/stl-a-seasonal-trend-decomposition-procedure-based-on-loess.pdf.*

19. *R version 4.0.2, "R: The R Project for Statistical Computing," 2020. https://www.r-project.org/ (accessed Aug. 25, 2020).*

20. *C. Bergmeir, R. J. Hyndman, and J. M. Benítez, "Bagging exponential smoothing methods using STL decomposition and Box–Cox transformation," Int. J. Forecast., vol. 32, no. 2, pp. 303–312, Apr. 2016, doi: 10.1016/J.IJFORECAST.2015.07.002.*

21. *A. Mishra, P. Chandra, U. Ghose, and S. S. Sodhi, "Bi-modal derivative adaptive activation function sigmoidal feedforward artificial neural networks," Appl. Soft Comput. J., vol. 61, pp. 983–994, Dec. 2017, doi: 10.1016/j.asoc.2017.09.002.*

22. *Pallab Kumar Datta, "An artificial neural network approach for short-term wind speed forecast," KANSAS STATE UNIVERSITY, 2018.*

23. *C. Bergmeir, R. J. Hyndman, and B. Koo, "A note on the validity of cross-validation for evaluating autoregressive time series prediction," Comput. Stat. Data Anal., vol. 120, pp. 70–83, Apr. 2018, doi: 10.1016/J.CSDA.2017.11.003.*

24. *M. Khairalla and N. T. Al-Jallad, "Hybrid Forecasting Scheme for Financial Time-Series Data using Neural Network and Statistical Methods," 2017. Accessed: Aug. 25, 2020. [Online]. Available: www.ijacsa.thesai.org.*