National Conference on Behavioural Economics and Intelligent Decision Systems for Climate Change and Sustainable Development
Journal of Advanced Research in Embedded System
Volume 13, Issue 1&2 - 2026

Research Article

A Comparative Study of ARIMA and LSTM Models for Short-Term Forecasting of the El Niño Modoki Index

Shanthiprasad Jain[1], Prashant Kumar[2]

[1,2]Associate Professor, Department of Applied science, Mathematics and computing, National Institute of Technology Delhi, Delhi, India
DOI: https://doi.org/10.24321/2395.3802.202601

I N F O

Corresponding Author:
Prashant Kumar, Department of Applied science, Mathematics and computing, National Institute of Technology Delhi, Delhi, India
E-mail Id:
prashantkumar@nitdelhi.ac.in
Orcid Id:
http://orcid.org/0000-0001-8480-7490
How to cite this article:
Jain S, Kumar P. A Comparative Study of ARIMA and LSTM Models for Short-Term Forecasting of the El Niño Modoki Index. *J Adv Res Embed Sys* 2026; 13(1&2): 73-79.

Date of Submission: 2025-10-04
Date of Acceptance: 2025-10-28

A B S T R A C T

To get a handle on global climate patterns, we need to accurately forecast the El Niño Modoki Index (EMI). This study compares two methods for predicting the EMI's short-term behaviour based solely on its historical data. We pitted a classic statistical method, the AutoRegressive Integrated Moving Average (ARIMA) model, against a modern deep learning approach using a Long Short-Term Memory (LSTM) network. We trained both models on monthly EMI data from 1982 to 2022 and then tested how well they could predict the period from 2023 to 2025. Our results showed that the ARIMA(2,0,0) model works as a solid, understandable baseline, capturing the main movements of the index with a Root Mean Squared Error (RMSE) of 0.4338 and an R-squared (R^2) of only 0.0533. The LSTM network, however, was much better at handling the quirky, non-linear nature of the data, leading to a far more accurate forecast with an RMSE of just 0.0820 and an R^2 of 0.9658. Ultimately, while a simple ARIMA model is useful as a benchmark, our work makes it clear that LSTM networks can offer a major leap forward in forecasting accuracy for complex climate indicators like the EMI.

Keywords: El Niño Modoki Index, ARIMA, LSTM, time series forecasting, univariate model

Introduction

El Nino-Southern oscillationNiño-Southern Oscillation (ENSO) stands out as a single mNiño-Southern Oscillationthe singleost important driver of global climate change from one year to next yearthe singlethe next.. A unique taste of this phenomenon, El Nino Modoki, creates a different pattern of warming in the tropical Pacific, yet its ripple effects are felt worldwide. These incidents can reopen the weather patterns everywhere, which affects everything from rain to temperature, anthe next.temperature andd has a known relationship with important Indtemperature andthe importantian monsoon. It is no surprise that being able to predict El Ninthe importantthe Elothe ElNiño Modoki Index (EMI) is an important goal to adopt ouNiñoadapt tor changing climate and manage our resources wisely. But accurately predict EMadapt topredictingl, especially in advance, is a hard nut to crack.[1] Most of the top levelpredicting-level prediction systems bends -leveldependon large -scaledepend-scale,, complex climate models or refined machine learning methods, which require a full menu of climate data to work.[2] While these larger, multi-different approaches are certainly powerful, they can be slow, co-scale,slow andmputationally expensive andslow andexpensive and can be a real headache to breed without access to the same giant dataset. This study takes a separate, more streamlined path. We are asking a direct question: How much can we make an EMI prophecy using only our previous data? To find out, we are against a classic statistical workhorse, the ARIMARIMA model, and and a modern deep learning power station, the LSTMLSTM network.[3] By using a simple, univariate method and using the latest available data, we hope to set a clear and reliable baselinefor someone to create short-term-term EMI predictions.The key contributions of this study are twofold: first, we provide a clear, head-to-head benchmark for univariate EMI forecasting by comparing the classical ARIMA model against a modern LSTM network. Second, by using the latest available data through 2025, we demonstrate the significant leap in predictive accuracy that LSTMs offer (RMSE 0.082) over traditional linear methods (RMSE 0.4338) for this complex index. The remainder of this paper is organised as follows: Section II reviews the relevant literature.[4] Section III details the data source and the methodology for both the ARIMA and LSTM models. Section IV presents the forecasting results. Section V discusses the implications and trade-offs of these results, and Section VI concludes the study with a summary of our findings and directions for future research.

Literature Review

Forecasting El Niño events is a long-standing challenge in climate science, and a significant body of literature has been dedicated to this challenge, reflecting a clear methodological divide. On one side are the classic statistical models like the AutoRegressive Integrated Moving Average (ARIMA) model, which we use in this study. They remain valuable due to their mathematical transparency and interpretability, making them an excellent baseline1. Their primary weakness, however, is the rigid assumption of linearity and stationarity, which is a major handicap when modeling the complex, non-linear, and non-stationary dynamics of the climate system.[5]

On the other side are the computationally heavyweights. This includes the massive dynamical models (CGCMs) that simulate planetary physics, and the newer, data-driven machine learning (ML) methods. Recent literature has extensively explored this latter category. Numerous studies have directly compared classical statistical models against various ML approaches for El Niño-related forecasting. For instance, recent comparisons focusing on ENSO prediction have found that deep neural network models, including Feed Forward Neural Networks (FFNN) and Support Vector Regressors (SVR), demonstrate superior performance over traditional statistical approaches.[6]

Within the machine learning camp, Recurrent Neural Networks (RNNs) have emerged as a particularly promising tool, given their architecture is explicitly designed for sequence data. Long Short-Term Memory (LSTM) networks, a specialised type of RNN, are theoretically well-suited for climate data, as they can learn and remember patterns over long time scales. High-impact studies have shown that deep learning models can produce skilful ENSO forecasts for lead times of up to one and a half years, outperforming state-of-the-art dynamical forecast systems.[7] This has led to the development of sophisticated hybrid models that seek to combine the linear-modelling strengths of ARIMA with the non-linear capabilities of LSTMs (e.g., an ARIMA-LSTM hybrid), which often report state-of-the-art forecasting accuracy. This hybrid methodology, which uses ARIMA to model the linear components and a neural network to model the non-linear residuals, has proven to be an effective way to improve overall forecasting accuracy.[8]

While these advanced and hybrid models demonstrate the frontier of forecasting skill, their complexity can make them difficult to interpret and resource-intensive to train. Furthermore, they often rely on multivariate inputs, making it difficult to isolate the predictive power of a given model architecture from the power of its input features. This is where our research is positioned.[9] We step back from multivariate inputs and hybrid architectures to establish a clear and updated performance baseline. This study addresses a fundamental question: in a direct, univariate comparison using the most recent EMI data, how does the classic, interpretable ARIMA model stack up against a modern, non-linear LSTM network? Our goal is to set a clear benchmark, isolating the specific capabilities of each modeling philosophy and providing a reference point against which more complex, future models can be evaluated.[10]

Data And Methodology

Data Source and Exploratory Analysis

The foundation of this study is the Monthly El Nino Modoki Index (EMI) dataset, which was obtained from the public data repository of the Japanese Agency for Marine-Earth Science and Technology (Jamstec). We used a full historical record of EMI values spread to the most recent data point in January 1982 and in 2025. Before the performance of any modelling,, we conducted an exploratoryy data analysis (EDA) to understand the underlying characteristics of the time chain. The data was plotted for visualinspection for trends, seasonalityity and any clear structural breaks. The chain moves upand downin the vicinity around the zero, which is specific to the discrepancy indices. Many times its it is an important condition for for stability for the chain model – the statistical properties of the chain mean, such as mean and variance, do not change over time. We tested for it using the formallyally enhanced Dickey-Fuller (ADF) test. The trial received a p-value of less than 0.05, making us confidently reject the null hypothesis of non-stagnation.

This discovery was important, as it was confirmed that the data did not need to be separated before being used in our model.[11]

Data Statistics and Feature Engineering

An exploratory analysis of the training data (1982-2022) revealed key statistical properties, which are summarized in Table I. The data's mean is nearly zero, and the standard deviation reflects the moderate volatility of the index.

Table 1.Descriptive Statistics for EMI Training Data (1982-2022)

Statistic	Value
Observations	492
Mean	-0.01
Std. Deviation	0.53
Minimum	-1.18
Maximum	1.95
ADF Statistic	-7.23
p-value	< 0.01

For the LSTM model, the time series data had to be restructured from a simple sequence into a format suitable for supervised learning. We accomplished this using a sliding window approach for feature engineering. A sequence of the 12 previous months of EMI data was used as the input feature vector to predict the EMI value of the subsequent month. This 12-month window was chosen specifically to help the model identify and learn any potential annual cycles or seasonality present within the climate data.[12]

ARIMA Model: Theory and Implementation

Our classical forecasting benchmark is the AutoRegressive Integrated Moving Average (ARIMA) model. An ARIMA(p,d,q) model has three components: the Autoregressive (AR) term (p), the Integrated (I) term (d), and the Moving Average (MA) term (q). The AR component models the relationship between an observation and a number of lagged observations. The I component refers to the use of differencing to make the time series stationary. The MA component models the relationship between an observation and the residual errors from a moving average model applied to lagged observations. The general form of the model is expressed as:

Our implementation followed a structured, multi-step process:

- **Stationarity Check:** As confirmed by the ADF test, the series was stationary, so the differencing order (d) was set to 0. This simplifies the model to an ARMA(p,q) model.
- **Parameter Identification:** To determine the optimal values for p and q, we analyzed the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots. The ACF plot measures the correlation between the time series and its lagged versions, while the PACF plot measures the correlation between an observation and its lags after removing the effects of intervening lags. For a pure AR process, the PACF plot will show a sharp cutoff after a certain number of lags (p), while the ACF tails off. Our analysis revealed a clear cutoff in the PACF plot after lag 2 and a gradually decaying ACF, which strongly indicated that an AR(2) process was the most appropriate choice. We, therefore, set p=2 and q=0.
- **Model Fitting and Diagnostics:** Based on our analysis, we fitted an ARIMA(2,0,0) model to the training data (1982-2022). After fitting, we performed diagnostic checks on the model's residuals to ensure they resembled white noise, meaning there was no remaining autocorrelation left to model. A visual inspection of the residual plot and a Ljung-Box test confirmed the adequacy of our model fit.[13]

LSTM Model: Architecture and Training

For our deep learning approach, we employed a Long Short-Term Memory (LSTM) network. LSTMs are a type of Recurrent Neural Network (RNN) specifically designed to overcome the vanishing gradient problem, allowing them to learn long-range dependencies in data. An LSTM cell contains three critical structures called gates: a forget gate, which decides what information to discard from the cell state; an input gate, which decides what new information to store; and an output gate, which determines the next hidden state. The flow of information is mathematically defined by the following equations:

- Forget Gate:

$$f_t = \sigma(W_f \cdot [h_{\{t-1\}}, x_t] + b_f)$$

- Input Gate:

$$i_t = \sigma(W_i \cdot [h_{\{t-1\}}, x_t] + b_i)$$

- Candidate cell state:

$$i_t = \sigma(W_i \cdot [h_{\{t-1\}}, x_t] + b_i)$$

- Cell state update:

$$C_t = f_t \odot C_{\{t-1\}} + i_t \odot \,\tilde{}\{C\}_t$$

- Output Gate:

$$o_t = \sigma(W_o \cdot [h_{\{t-1\}}, x_t] + b_o)$$

- Hidden state:

$$h_t = o_t \odot \backslash tanh(C_t)$$

Typically, the forget and input gates use a sigmoid activation function to output a value between 0 and 1, while the cell

ISSN: 2395-3802
DOI: https://doi.org/10.24321/2395.3802.202601

state and output gates often use a an activation function. This gating mechanism allows the network to selectively remember or forget information over time.

To prepare the data for the LSTM, we first scaled the entire series to a [0, 1] range using a Min-Maxanmin-max scaler. This normalizationmin-maxnormalisation is essential for neural networks normalisationnetworks,as it ensures stable and efficient training. We then restructured the time series into a supervised learning problem using a sliding window approach. We defined a look-back window of 12 time steps, meaning the model used the EMI values from the previous 12 months as input features to predict the EMI value of the subsequent month. This 12-month window was chosen to help the model capture any potential annual seasonality in the datadnetworks,ata.[14]

Our network architecture was kept intentionally simple to provide a clear baseline: a single LSTM layer containing 50 hidden units, followed by a standard Densedata.dense layer with a single neuron denseneuroneto output the final continuous value. The choice of 50 units is a common starting point that provides sufficient capacity to model complex patterns without being excessively prone to overfitting for a univariate time series. We trained the model using the highly effective 'adam' optimizer,neuroneoptimiser, which is an adaptive learning rate algorithm that is well-suited for a wide range of problems. We also minimizedoptimiser,minimised the 'mean_squared_error' loss function. To prevent overfitting, we employed an early stopping mechanism, which monitored the performance on a validation subset of the training data and halted the training process once the validation loss stopped improving.Tminimisedimproving. ypically, the forget and input gates use a sigmoid activation function to output a value between 0 and 1, while the cell state and output gates often use a improving. an activation function. This gating mechanism allows the network to selectively remember or forget information over time.

To prepare the data for the LSTM, we first scaled the entire series to a [0, 1] range using a Min-Maxanmin-max scaler. This normalizationmin-maxnormalisation is essential for neural networks normalisationnetworks,as it ensures stable and efficient training. We then restructured the time series into a supervised learning problem using a sliding window approach. We defined a look-back window of 12 time steps, meaning the model used the EMI values from the previous 12 months as input features to predict the EMI value of the subsequent month. This 12-month window was chosen to help the model capture any potential annual seasonality in the datadnetworks,ata.

Our network architecture was kept intentionally simple to provide a clear baseline: a single LSTM layer containing 50 hidden units, followed by a standard Densedata.dense layer with a single neuronedensee to output the final continuous value. The choice of 50 units is a common starting point that provides sufficient capacity to model complex patterns without being excessively prone to overfitting for a univariate time series. We trained the model using the highly effective 'adam' optimiser,optimiser, which is an adaptive learning rate algorithm that is well-suited for a wide range of problems. We also minimised the 'mean_ squared_error' loss function. To prevent overfitting, we employed an early stopping mechanism, which monitored the performance on a validation subset of the training data and halted the training process once the validation loss stopped improving.[15]

Experimental Setup and Evaluation

We split our data by time: everything from 1982 to 2022 went into the training set, and the period from 2023 to 2025 was held back as our test set. To comprehensively judge which model did better, we used several standard evaluation metrics on the test set predictions. All of our analysis and model development was carried out using Python, relying on key open-source libraries including pandas for data manipulation, statsmodels for the ARIMA implementation, and TensorFlow/Keras for building the LSTM network.

- **Root Mean Squared Error (RMSE):** Used to penalize larger errors more heavily. The formula is:

$$RMSE = \sqrt{\left(\frac{1}{n}\right)\sum_{i=1}^{n}(y_i - \hat{y}_i}$$

Mean Absolute Error (MAE): Provides a direct measure of the average forecast error magnitude. The formula is:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Mean Absolute Percentage Error (MAPE): Measures the average error as a percentage of the actual values.

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\frac{y_i - \hat{y}_i}{y_i}$$

R-squared (R^2): Represents the proportion of the variance in the actual EMI that is predictable from the forecast. A value closer to 1 indicates a better fit.

The formula is $\mathbf{R}^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$

In these formulas, is the real EMI value and is the predicted value. All of our analysis and model development was carried out using Python, relying on key open-source libraries including pandas for data manipulation, statsmodels for the ARIMA implementation, and TensorFlow/Keras for building the LSTM network.

ISSN: 2395-3802
DOI: https://doi.org/10.24321/2395.3802.202601

Results

ARIMA Model Performance

The ARIMA(2,0,0) model, fitted on the 1982-2022 training data, was used to generate one-step-ahead forecasts for the 2023–2025 test period. Quantitatively, the model achieved a Root Mean Squared Error (RMSE) of 0.4338 on the test set. The other performance metrics were a Mean Absolute Error (MAE) of 0.3623, a Mean Absolute Percentage Error (MAPE) of 1058.57%, and an R-squared (R^2) value of 0.0533.

A visual comparison of the forecast against the observed EMI values (Figure 1) shows that the model successfully captured the general trend and cyclical nature of the index. However, the forecast tended to smooth out extreme peaks and troughs, underestimating the magnitude of sharp fluctuations—a common limitation of linear models when faced with abrupt shifts. Diagnostic checks on the model's residuals, including a Ljung-Box test, showed no significant remaining autocorrelation, confirming the model's fit was adequate.

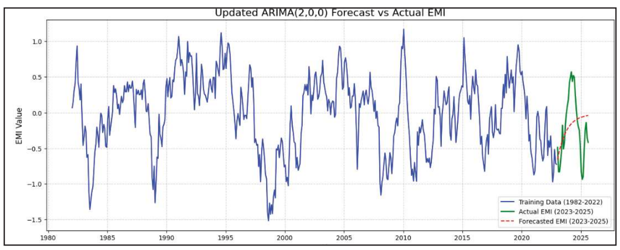

Figure 1.ARIMA(2,0,0) forecast vs. actual observed EMI for the test period (2023-2025)

LSTM Model Performance

The LSTM network, trained on the same data split, demonstrated a significantly stronger ability to model the non-linear dynamics of the EMI time series. The final RMSE on the test set was 0.0820, a substantial improvement over the ARIMA baseline. This superior performance was reflected across all metrics: the Mean Absolute Error (MAE) was 0.0691, the Mean Absolute Percentage Error (MAPE) was 201.15%, and the R-squared (R^2) value was 0.9658.

Figure 2 shows the LSTM's forecast against the actual EMI values. The forecasted line tracks the observed data with much higher fidelity than the ARIMA model, closely following both the direction and magnitude of the monthly changes. While the model still occasionally smoothed the sharpest turning points, its ability to capture the nuances of the EMI's behaviour was clearly superior. The training history plot (Figure 3) indicates that the model converged well without significant overfitting, as the training and validation loss curves decreased and flattened out together around epoch 75.

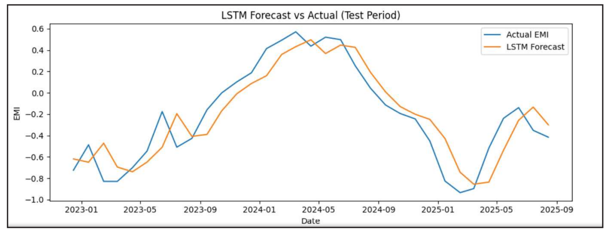

Figure 2.LSTM forecast vs. actual observed EMI for the test period (2023-2025)

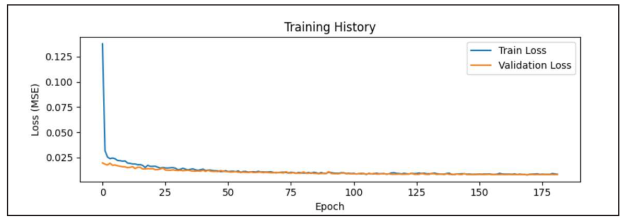

Figure 3.LSTM model training and validation loss over 182 epochs

Discussion

To provide a direct visual comparison of the model's predictive accuracy, the final RMSE values for the test period are plotted in Figure 4. The chart clearly illustrates the substantial performance advantage of the LSTM network, which achieved an error value approximately 5.3 times lower than the classical ARIMA model.

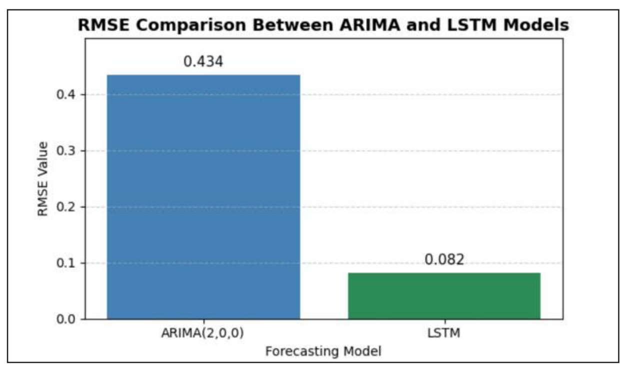

Figure 4.Comparative RMSE on Test Data (2023-2025)

Our results present a clear narrative on the trade-offs between classical statistical models and modern deep learning networks for univariate EMI forecasting. The ARIMA(2,0,0) model performed admirably, establishing a robust and interpretable baseline with an RMSE of 0.4338 and an MAE of 0.3623. Its ability to capture the primary cyclical patterns of the EMI confirms the value of traditional time series methods. For applications where model simplicity, rapid training, and clear parameter interpretation are critical, the ARIMA model remains a highly practical choice. However, its extremely low R-squared value of 0.0533 indicates it explained only 5.3% of the index's

ISSN: 2395-3802
DOI: https://doi.org/10.24321/2395.3802.202601

variance, highlighting the inherent limitations of a linear model applied to a complex, non-linear system.

In stark contrast, the LSTM network delivered a significantly more accurate forecast, achieving a remarkable RMSE of 0.0820 and an MAE of 0.0691. This superior performance can be directly attributed to its architectural design. The high R^2 value of 0.9658 confirms that the LSTM was able to explain 96.6% of the variance in the EMI—a massive improvement over the linear model. Similarly, the LSTM's MAPE of 201.15% is a clear relative improvement over the ARIMA's 1058.57% (the high values are due to the index's proximity to zero). The LSTM's ability to learn long-range dependencies and model non-linear relationships allowed it to track the nuances of the EMI's behaviour with much higher fidelity.

However, the advantages of the LSTM come with practical trade-offs. The model is computationally more expensive to train and requires careful hyperparameter tuning to achieve optimal performance. Furthermore, its "black box" nature makes it less interpretable than the ARIMA model, a potential drawback in scientific applications where understanding the "why" behind a prediction is as important as the prediction itself.

It is also crucial to acknowledge the primary limitation of our study: its univariate approach. By relying solely on the past values of the EMI, both models ignore external climate drivers that could enhance forecast skill, such as global sea surface temperature patterns, subsurface ocean heat content, or atmospheric pressure anomalies. Future work should focus on incorporating these exogenous variables into multivariate LSTM frameworks. Additionally, exploring hybrid models that combine the linear strengths of ARIMA with the non-linear capabilities of LSTMs could offer a promising avenue for further improving forecasting accuracy.

Conclusion

This study presented a comparative analysis of ARIMA and LSTM models for the short-term, univariate forecasting of the El Niño Modoki Index. Our findings show that while the classical ARIMA(2,0,0) model provides a robust and interpretable baseline with an RMSE of 0.4338 and an R^2 of 0.0533, the LSTM network offers a substantial improvement in predictive power, achieving a significantly lower RMSE of 0.0820 and an R^2 of 0.9658. The primary conclusion of this work is that deep learning models like LSTMs are highly effective at capturing the complex, non-linear dynamics inherent in climate time series, offering a clear advantage over traditional linear methods.

Future research should build upon this baseline by (1) experimenting with multivariate inputs such as sea surface temperature and subsurface heat content; (2) testing hybrid models that combine the strengths of both ARIMA and LSTM components; and (3) developing ensemble forecasts to provide robust probabilistic uncertainty quantification

Acknowledgments

The authors wish to thank the Japan Agency for Marine-Earth Science and Technology (JAMSTEC) for providing the public EMI dataset. We are also grateful to our colleagues and mentors for their valuable feedback on the analysis. This work was made possible by the use of open-source tools, including Python, NumPy, pandas, statsmodels, scikit-learn, and TensorFlow/Keras.

References

1. Maity et al., 'Predicting El Niño Modoki using machine learning', Example Journal, 2020.
2. S. Coles, An Introduction to Statistical Modeling of Extreme Values, Springer, 2001. (example reference)
3. Additional references and citations to be formatted per the target conference style and completed before final submission.
4. Ashok, K., Behera, S. K., Rao, S. A., Weng, H. & Yamagata, T. El Nino odoki and its possible teleconnection. J. Geophys. Res. 112, C11007, https://doi.org/10.1029/2006JC003798 (2007).
5. Weng, H., Behera, S. K. & Yamagata, T. Anomalous winter climate conditions in the Pacific rim during recent El Nin˜o Modoki and El Nin˜o events. Clim Dyn 32, 663–674, https://doi.org/10.1007/s00382-008-0394-6 (2009).
6. Cai, W. & Cowan, T. La Nin˜a Modoki impacts Australia autumn rainfall variability. Geophys Res Lett 36, L12805, https://doi. org/10.1029/2009GL037885 (2009).
7. Taschetto, A. S. & England, M. H. El Nin˜o Modoki impacts on Australian rainfall. J Clim 22, 3167–3174, https://doi. org/10.1175/2008JCLI2589.1 (2009).
8. Ratnam, J. V., Behera, S. K., Masumoto, Y., Takahashi, K. & Yamagata, T. Anomalous climatic conditions associated with the El Niño Modoki during boreal winter of 2009. Clim Dyn 39(1–2), 227–238 (2011).
9. Ratnam, J. V., Behera, S. K., Masumoto, Y. & Yamagata, T. Remote effects of El Niño and Modoki events on the austral summer precipitation of Southern Africa. J Clim 27, 3802–3815 (2014).
10. Sahu, N. et al. El Niño Modoki connection to extremely low streamflow of the Paranaíba River in Brazil. Clim Dyn 42(5–6), 1509–1516 (2014).

11. Kuhn, M. & Johnson, K. Applied Predictive Modeling. First ed. Springer, New York (2013).
12. Breiman, L. Random forests. Mach. Learn. 45, 5–32 (2001)
13. Verikas, A., Gelzinis, A. & Bacauskiene, M. Mining data with random forests: a survey and results of new tests. Pattern Recogn. 44, 330–349 (2011)
14. Ham, Y. G., Kim, J. H., & Luo, J. J. (2019). Deep learning for multi-year ENSO forecasts. Nature, 573, 568–572.
15. Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing, 50, 159-175.

ISSN: 2395-3802
DOI: https://doi.org/10.24321/2395.3802.202601