AJAS / SPAS

African Journal of Applied Statistics
Vol. 12 (1), 2025, pages 1711 - 1730.
DOI: https://dx.doi.org/ajas/2025.1711.289

ISSN 2316-0861

Temporal Variability and LSTM Forecasting of Ugandan Time Series

Gilbert Mutungi [(1,*)], **Ronald Wesonga** [(2)], **Francis Bbosa** [(1)] **and Richard Tuyiragize** [1]

[(1)] School of Statistics and Planning, Makerere University, Uganda
[*2)] Department of Statistics, Sultan Qaboos University, Oman

Received on September 9, 2026; Accepted on October 01, 2026; Published online November 13, 2026

Copyright © 2025, The African Journal of Applied Statistics (AJAS) and The Statistics and Probability African Society (SPAS). All rights reserved

Abstract. The performance of Long Short-Term Memory (LSTM) networks in time series forecasting is often assumed to be robust, yet their sensitivity to different types of temporal variations remains underexplored. We investigate the performance of LSTMs on three Ugandan and one M4 time series with distinct patterns. Forecast performance is assessed in terms of accuracy and uncertainty, measured respectively by Mean Absolute Percentage Error (MAPE) and the Coefficient of Variation (CV) of forecasts across repeated independent model runs. Results show that high lumpiness significantly degrades LSTM accuracy and certainty, often making ARIMA a more reliable baseline. However, applying appropriate transformations substantially improves LSTM performance on lumpy series, reducing MAPE by up to 35%. Further, the models achieve their best performance on series with a single dominant structural pattern, either trend or seasonality. These findings highlight the importance of understanding data structure when applying LSTM models to time series forecasting.

Key words: LSTM; lumpiness; forecasting; temporal variability
AMS 2010 Mathematics Subject Classification Objects : G62P05; 62M10; 62M20.

[(*)] Corresponding author: Gilbert Mutungi (gilbertmutungi@gmail.com)
Ronald Wesonga : wesonga@wesonga.com
Francis Bbosa : fullerbbosa@gmail.com
Richard Tuyiragize : trgzrich@yahoo.com

Résumé (Full Abstract in French) Les performances des réseaux Long Short-Term Memory (LSTM) en prévision des séries chronologiques sont souvent considérées comme robustes, mais leur sensibilité aux différents types de variations temporelles reste encore peu étudiée. Nous analysons ici les performances des LSTM sur trois séries ougandaises et une série M4 présentant des structures distinctes. Les performances prévisionnelles sont évaluées en termes de précision et d'incertitude, mesurées respectivement par l'erreur absolue moyenne en pourcentage (MAPE) et le coefficient de variation (CV) des prévisions obtenues lors d'exécutions indépendantes répétées du modèle. Les résultats montrent qu'une forte irrégularité ("lumpiness") dégrade sensiblement la précision et la stabilité des LSTM, rendant souvent ARIMA plus fiable comme modèle de référence. Toutefois, des transformations appropriées améliorent nettement les performances des LSTM sur les séries irrégulières, avec une réduction de la MAPE pouvant atteindre 35 %. Les meilleures performances sont obtenues pour les séries présentant une structure dominante unique, tendance ou saisonnalité. Ces résultats soulignent l'importance de bien comprendre la structure des données avant d'appliquer les modèles LSTM à la prévision des séries chronologiques.

Presentation of the authors.

Gilbert Mutungi, M.Sc., is an Assistant Lecturer at : School of Statistics and Planning, Makerere University, Uganda. He is preparing a PhD thesis in Statistics under the supervision of the second and the fourth authors and Prof Johan Lyhagen

Ronald Wesonga, Ph.D., is an Associate Professor of Statistics at the Department of Statistics, Sultan Qaboos University, Oman.
Francis Bbosa, Ph.D, is He is a Lecturer at the School of Statistics and Planning, Makerere University, Uganda.
Richard Tuyiragize, Ph.D, is a Lecturer at the School of Statistics and Planning, Makerere University, Uganda.

1. Introduction

The theory of forecasting is grounded in the idea that the future carries echoes of the past. Quantitative forecasting, in particular, rests on the fact that historical time series exhibit temporal variations that can be exploited to form conditional predictions. Accordingly, the accuracy of forecasting methods depends on how they adapt to, or capture, these underlying patterns. The most commonly used forecasting methods can be broadly categorized into model-driven and data-driven methods.

Model-driven methods rely on the statistical properties of historical observations to specify a formal model and estimate its parameters. While effective for series with a relatively stable temporal structure, these approaches face significant challenges in the presence of high and irregular temporal variability. In such cases,

it becomes difficult to specify a stochastic framework that adequately captures the observed dynamic behavior. This limitation gives rise to data-driven methods. Rather than imposing an explicit model, these aim to learn temporal structure directly from the observed data. No prior relations need to be assumed or estimated by the forecaster. Among these methods, gated recurrent neural networks—and in particular Long Short-Term Memory (LSTM) networks—have emerged as some of the most popular and successful tools for time series forecasting. The gating mechanisms in LSTMs enable selective retention and forgetting of information, allowing relevant temporal signals to be preserved over long sequences.

Insights from the seminal work of Hochreiter and Schmidhuber (1997) suggest that LSTMs are robust to noisy signals, irrelevant inputs, and sequences where informative events occur at variable and unpredictable time points. Although these properties were not originally framed in terms of temporal variability, subsequent studies have extended these ideas to more complex and irregular time series. For example, Mehedi *et al.* (2022) demonstrates that LSTMs can effectively model chaotic and dynamic variations in river discharge, while Giannopoulos *et al.* (2025) identify LSTMs as among the preferred recurrent architectures for lumpy and intermittent demand patterns.

At the same time, growing evidence suggests that high levels of temporal variability can adversely affect the forecasting performance of deep learning models, including LSTMs. Studies such as Singh *et al.* (2024) report deteriorating performance as rainfall variability increases, while Shafi *et al.* (2023) shows that LSTM accuracy declines in the presence of pronounced lumpiness. Most of these studies focus on comparing the model with other architectures for a given time series, rather than investigating how the model performs across series with different temporal variations. As a result, the question of the performance of LSTMs on various types of temporal variations remains open.

This study addresses this gap by comparing LSTM forecast performance across four time series with distinct temporal patterns. First, we evaluate the uncertainty and accuracy of LSTM forecasts across multiple independent training runs for each series. Second, we investigate whether transformations designed to reduce temporal variability improve both forecast certainty and accuracy.

Our results show that the model exhibits high levels of uncertainty for time series with pronounced lumpiness and with both trend and seasonal patterns present. This is reflected in the high variability of forecasts across independent runs for such series. When variability-reducing transformations are applied, both forecast variability and accuracy improve substantially.

However, we also observe that the model performs well when a series is dominated by a single strong and homogeneous trend or seasonal pattern, but struggles when both are present simultaneously.

The remainder of the paper is organized as follows. Section 2 presents the LSTM architecture. Section 3 discusses temporal variation and its components while Section 4 presents the data. Section 5 outlines the experimental design and evaluation metrics. Section 6 discusses the results, and Section 7 concludes the paper.

2. Long Short-Term Memory Networks

In this section, we provide a detailed description of the LSTM model.

2.1. LSTM Description

The Long Short-Term Memory (LSTM) network is an advanced form of recurrent neural network (RNN) designed to mitigate the vanishing gradient problem, thereby enabling more effective learning of long-term dependencies (Gers *et al.* , 2000). It retains information over time through specialized memory cells and a gating mechanism that regulates information flow. Each memory block contains three gates: the forget gate f_t, the input gate i_t, and the output gate o_t. At each time step t, these gates operate on the current input x_t and the hidden state from the previous time step h_{t-1}, while the cell state c_t acts as the internal memory that carries information through the sequence.

The *forget gate* determines which portions of the previous memory c_{t-1} to retain or discard. It outputs a value between 0 and 1, where 1 signifies complete retention, and 0 signifies complete forgetting. The *input gate* regulates the extent to which new information updates the current cell state. This gate considers both the current input and the previous hidden state. The *cell candidate* g_t determines which new data should be integrated into the cell state. It uses a $\tanh$ activation function, which helps maintain values in the $[-1, 1]$ range and facilitates smoother gradient flow. The previous cell state c_{t-1} and the candidate information g_t are then combined, according to the forget and input gates, respectively, to produce the updated cell state c_t. The *output gate* determines how much of the internal cell state should be passed on to the next layer or as output. Finally, the hidden state h_t is computed by applying the $\tanh$ activation to the updated cell state, scaled by the output gate. These operations are summarised in the equations below:

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right), \tag{1}$$
$$i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + b_i\right), \tag{2}$$
$$g_t = \tanh\left(W_{xg}x_t + W_{hg}h_{t-1} + b_g\right), \tag{3}$$
$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + b_o\right), \tag{4}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot g_t, \tag{5}$$
$$h_t = o_t \odot \tanh(c_t), \tag{6}$$

where x_t is the input vector at time t, h_{t-1} and h_t are the hidden states at the previous and current time steps, respectively, and c_{t-1} and c_t are the corre-

sponding previous and current cell states. The quantities f_t, i_t, and o_t denote the forget, input, and output gate vectors, respectively, while g_t denotes the candidate cell-state vector. Weight matrices W and biases b are parameters learned during training. More specifically, W_x represents the weights connecting the current input x_t to the corresponding gate or cell candidate, whereas W_h represents the weights connecting the previous hidden state h_{t-1}; the subscripts f, i, g, and o identify the forget gate, input gate, cell candidate, and output gate, respectively. The terms b_f, b_i, b_g, and b_o are their corresponding bias vectors. The functions σ and $\tanh$ denote the sigmoid and hyperbolic tangent activations, respectively. The sigmoid function maps values to the interval $(0, 1)$, allowing the gates to regulate the proportion of information retained or transmitted, whereas the hyperbolic tangent function maps values to $(-1, 1)$. The operator $\odot$ denotes the Hadamard (element-wise) product, meaning that corresponding elements of the two vectors are multiplied.

Together, Equations (1)–(6) describe the LSTM update from one time step to the next. Given x_t, h_{t-1}, and c_{t-1}, the network first computes the forget gate f_t, input gate i_t, candidate information g_t, and output gate o_t. The new cell state c_t is then obtained by combining the retained portion of the previous cell state, $f_t \odot c_{t-1}$, with the selected candidate information, $i_t \odot g_t$. Finally, the updated cell state is transformed by $\tanh$ and regulated by the output gate to produce h_t. The resulting c_t and h_t are subsequently carried forward to the next time step, where the same sequence of operations is repeated.

2.2. LSTM training, convergence, and Uncertainty

Training an LSTM involves estimating the weights and biases associated with its gating mechanisms and state transitions. This is achieved through backpropagation through time (BPTT) (Werbos , 1990), which computes gradients of the loss function with respect to the model parameters across multiple time steps. These gradients are then used to update the parameters using optimization algorithms such as stochastic gradient descent (SGD), RMSprop, or Adam.

Once trained, the LSTM model, parameterized by the set of optimal parameters, θ provides an estimate of the conditional expectation

$$\hat{y}_{t+1|t} = E\big[\hat{y}_{t+1} \mid \hat{y}_t, \hat{y}_{t-1}, \ldots, \hat{y}_{t-p}\big] \tag{7}$$

where $\hat{y}_{t+1}$ denotes the point forecast and p is the selected input lag length.

As with other neural networks, LSTM training is a non-convex optimization problem. As a result, different independent training runs—initialized with different random seeds—may converge to distinct local optima, producing a different set of θ and, consequently, different forecasts. The level at which these vary reflects how certain the model is in its predictions. One common method of measuring this

uncertainty is by measuring the variance of predictions over several independent runs (Lakshminarayanan *et al.* , 2017). Given that we aim to compare this level of uncertainty over different datasets, we use the coefficient of variation.

3. Temporal Variability

In this section, we discuss temporal variation.

Temporal variability refers to the fluctuations, shifts, and changes that occur in a system over time. A time series may be characterized by a single dominant type of variation or by a combination of multiple forms of variability.

Our study focuses on variability due to trend, seasonal fluctuations, and lumpiness.

Variation due to trend

A trend is defined as a persistent increase or decrease in a time series. Hyndman and Athanasopoulos (2021) proposed a measure to quantify the proportion of variation attributable to the trend component. Their approach assesses trend strength by comparing the variance of the irregular component to the variance of the deseasonalised series. Intuitively, when a trend explains most of the variation in the data, the remaining irregular variance should be small.

Formally, given a time series Y_t, it can be decomposed into trend (T_t), seasonal (S_t), and irregular (I_t) components (Hyndman and Athanasopoulos , 2021). The strength of variation due to trend, denoted by T, is defined as

$$T = \max\left(0,\ 1 - \frac{\mathrm{Var}(I_t)}{\mathrm{Var}(Y_t - S_t)}\right) \tag{8}$$

This measure takes values between 0 and 1, with larger values indicating a stronger and more dominant trend component.

Seasonal strength

Seasonality refers to the repeated regular fluctuations in a time series often attributed to seasonal or calendar factors (Kang *et al.* , 2017). Following from the same intuition above, the variation attributed to seasonality, Seasonal Strength, S, is computed as

$$S = \max\left(0,\ 1 - \frac{\mathrm{Var}(I_t)}{\mathrm{Var}(Y_t - T_t)}\right) \tag{9}$$

The value ranges from 0 to 1, with higher values indicating a stronger seasonality.

Lumpiness

Lumpiness measures the extent to which variability changes across different segments of a time series. It is defined as the variability of segment-wise (tiled) variances, capturing heterogeneity in variance over time. We compute lumpiness using the `tsfeatures` (Hyndman *et al.* , 2026) package that uses the algorithm below:

1. Standardize the time series Y_t.
2. Partition the series into K consecutive, non-overlapping segments, each with length equal to the seasonal period.
3. For each segment $k = 1, \ldots, K$, compute the sample variance v_k.
4. Define lumpiness as

$$\text{Lumpiness} = \text{Var}(v_1, \ldots, v_K).$$

Unlike trend and seasonal strength, lumpiness reflects changes in local variability and can often be substantially reduced through an appropriate Box–Cox transformation.

4. Data

This section describes the datasets employed in the study, including their sources, structural characteristics, and pre-processing steps.

4.1. Dataset Description

Our analysis utilizes four univariate time series exhibiting diverse structural properties. Maize prices are selected as a staple commodity in Uganda, known to exhibit pronounced seasonal and trend components (Cedrez *et al.* , 2020). Rainfall volumes are included due to their well-documented seasonal variability (Nsubuga *et al.* , 2014), while the inflation series are chosen for its characteristic lumpiness and trending behavior (Kemoe *et al.* , 2024). The M42869 series, sourced from the M4 Forecasting Competition, is incorporated as a benchmark with notable lumpiness and complex patterns commonly used in forecasting research. Summary details of each series follow.

4.1.1. Maize Prices

This monthly series records the national average retail prices of white maize in Uganda spanning January 2014 to June 2024. The data are obtained from the Food Price Monitoring and Analysis Tool (FPMA) of the United Nations Food and Agriculture Organization (FAO) (Food and Agriculture Organization of the

United Nations , 2026). Prices range from 600 to 2,088 Ugandan shillings and exhibit a strong trend strength (0.91), moderate seasonal strength (0.51), and low lumpiness (0.11).

4.1.2. Rainfall Volumes

Rainfall data captures the volumes measured during the first decade of each month from January 2013 to October 2024 in Uganda. These observations were sourced from the World Food Programme's Data Viz platform (World Food Programme , 2026). Rainfall values range from 2.59 to 89.18 mm, displaying moderate seasonal strength (0.62), low trend strength (0.14), and lumpiness of 0.20.

4.1.3. Inflation

This monthly time series reports year-on-year inflation rates in Uganda between January 2009 and September 2024, sourced from the World Food Programme's Data Viz platform (World Food Programme , 2026). Inflation rates vary from 3.15% to 24.40%, characterized by a very high trend strength (0.93), minimal seasonal variation (0.02), and comparatively higher lumpiness (0.52).

4.1.4. M42869 Series

The M42869 series is a monthly financial time series from the M4 Forecasting Competition (Makridakis *et al.* , 2020), accessible via Kaggle (Makridakis *et al.* , 2018). It ranges from 124.4 to 2,551.63 units and exhibits dual patterning with seasonal strength (0.72), trend strength (0.69), and lumpiness of 0.41.

4.2. Data Transformation and Pre-processing

Each time series $\{x_t\}_{t=1}^{T}$ is split into training and test sets, with the final six observations reserved for out-of-sample evaluation.

To assess the effect of data transformation on forecasting performance, a Box–Cox transformation is applied to the training data:

$$x_t^{(\lambda)} = \begin{cases} \dfrac{x_t^{\lambda} - 1}{\lambda}, & \lambda \neq 0, \\ \log(x_t), & \lambda = 0, \end{cases} \tag{10}$$

where the transformation parameter λ is estimated using the Guerrero method, constrained to the interval $[0, 1]$. For $\lambda < 0$ or values close to zero, a log trans-

Dataset	Range	Mean	Seasonality	Trend	Lumpiness
Maize	126.8–307	198.2	0.518	0.908	0.076
Rainfall	1.059–7.72	5.211	0.720	0.160	0.180
Inflation YoY	0.405–3.19	1.450	0.029	0.909	0.086
M42869	4.82–9.26	7.840	0.820	0.670	0.063

Table 1. Structural features of the transformed training series

formation is used. First-order differencing is applied to all series identified as non-stationary by the Augmented Dickey–Fuller (ADF) test.

All pre-processing steps are performed exclusively on the training data to avoid information leakage. Finally, the forecasts are back-transformed to the original scale for accuracy evaluation. Table 1 summarizes the structural features of the transformed training series.

5. Experimental Setup

This section outlines the implementation and evaluation procedures for the LSTM models used in the study.

5.1. Model Implementation

5.1.1. LSTM Model Parameter Tuning

The LSTM models are implemented in R version 4.4.0 using TensorFlow 2.1.5 via the `tensorflow` R package (Allaire *et al.* , 2018). The architecture consists of two stacked LSTM layers followed by a dropout layer and a dense output layer with six units, corresponding to the six-step-ahead forecasting horizon. This structure reflects widely adopted configurations for time series forecasting in prior studies (Bala *et al.* , 2019; Rguibi *et al.* , 2022).

Hyperparameter tuning is performed using a 3-fold sliding-window cross-validation approach. Hidden unit sizes of 128 and 64 are considered for the first and second LSTM layers, respectively, alongside dropout rates of 0.2 and 0.3. The Adam optimizer, with default Keras settings, is employed for adaptive learning. Early stopping was applied based on validation loss to mitigate overfitting. The selected hyperparameters for each series are reported in Table A1. To account for variability due to random weight initialization, the final training and forecasting process was repeated 30 times. Although previous research suggests that 6 to 9 repetitions are sufficient to estimate forecast uncertainty (Smyl , 2020), the choice of 30 aligns with the Central Limit Theorem.

5.2. Performance Evaluation

Model performance is evaluated along two dimensions: model uncertainty and forecast accuracy.

Model Uncertainty

We quantify model uncertainty by computing the variation in LSTM forecasts across independent training runs. Accordingly, for each time series, model training and forecasting are repeated 30 times. The variability across these runs is then summarized using the Coefficient of Variation (CV) of the forecasts. While variance or standard deviation are commonly used for this purpose, we adopt the CV to allow comparison of uncertainty levels across different data sets. Further, the CV is computed separately at each forecast horizon to examine how this uncertainty evolves as the forecast horizon increases.

For a given forecast horizon i, the CV is defined as

$$\text{CV}_i = \frac{\sigma_i}{\mu_i} \times 100\%, \tag{11}$$

where σ_i and μ_i denote the standard deviation and mean, respectively, of the forecast values at horizon i across the 30 repetitions. This metric captures the relative variability of the forecasts.

Accuracy

Accuracy is computed using the Mean Absolute Percentage Error (MAPE).

MAPE for horizon i is computed as:

$$\text{MAPE}_i = \frac{1}{n} \sum_{j=1}^{n} \left| \frac{x_{i,j} - \hat{x}_{i,j}}{x_{i,j}} \right| \times 100\%, \tag{12}$$

where $x_{i,j}$ is the actual value, $\hat{x}_{i,j}$ is the predicted value at horizon i for the j-th instance, and n is the number of test observations.

Average MAPE per horizon was calculated as:

$$\text{AvgMAPE}_i = \frac{1}{30} \sum_{k=1}^{30} \text{MAPE}_{i,k}, \tag{13}$$

where $\text{MAPE}_{i,k}$ denotes the MAPE at horizon i for the k-th run. Finally, a single summary accuracy metric was obtained by computing the mean MAPE across all six forecast horizons:

$$\text{MeanMAPE} = \frac{1}{6}\sum_{i=1}^{6} \text{AvgMAPE}_i. \tag{14}$$

These metrics were used to compare the LSTM model's performance against the ARIMA benchmark.

6. Experimental Results and Discussion

This section presents and discusses the study's empirical findings.

6.1. Variability in LSTM forecasts

Figure 1 presents the coefficient of variation (CV) of LSTM forecasts across six horizons, comparing results before and after data transformation.

Overall, the coefficient of variation (CV) of LSTM forecasts increases with the forecast horizon. Among the raw series, M42869 and inflation, which initially exhibit the highest lumpiness, also show the highest CVs, indicating greater forecast variability. In contrast, raw maize price and rainfall series have relatively lower lumpiness and correspondingly lower CVs.

This result shows that high model uncertainty is more pronounced in time series with higher lumpiness. This also demonstrates that the uncertainty increases with an increase in the forecast horizon.

When the highly lumpy series are transformed, the CV not only decreases but also stabilizes across horizons. For the less lumpy series, transformations have little effect on either the magnitude or behavior of the CV. This result aligns with several findings that the model uncertainty in neural networks increases with increased input noise in datasets (Valdenegro-Toro *et al.* , 2024). Xu *et al.* (2021) explains that high input variability is translated through the model's training process in backpropagation. This leads to increased variation and more dispersive weights.

Further, it is important to note that, despite substantial reductions, the CV of the M42869 series remains high. In the transformed series, M42869 and maize prices continue to have the highest CVs. We attribute this to the presence of strong dual patterns — both trend and seasonality — which cannot be fully mitigated by Box-Cox transformations. In contrast, transformed inflation and rainfall series, with more homogeneous trend or seasonal patterns, show lower and more stable forecast variability.

6.2. Accuracy of LSTM Forecasts

Forecast accuracy across the six horizons is shown in Figure 2.

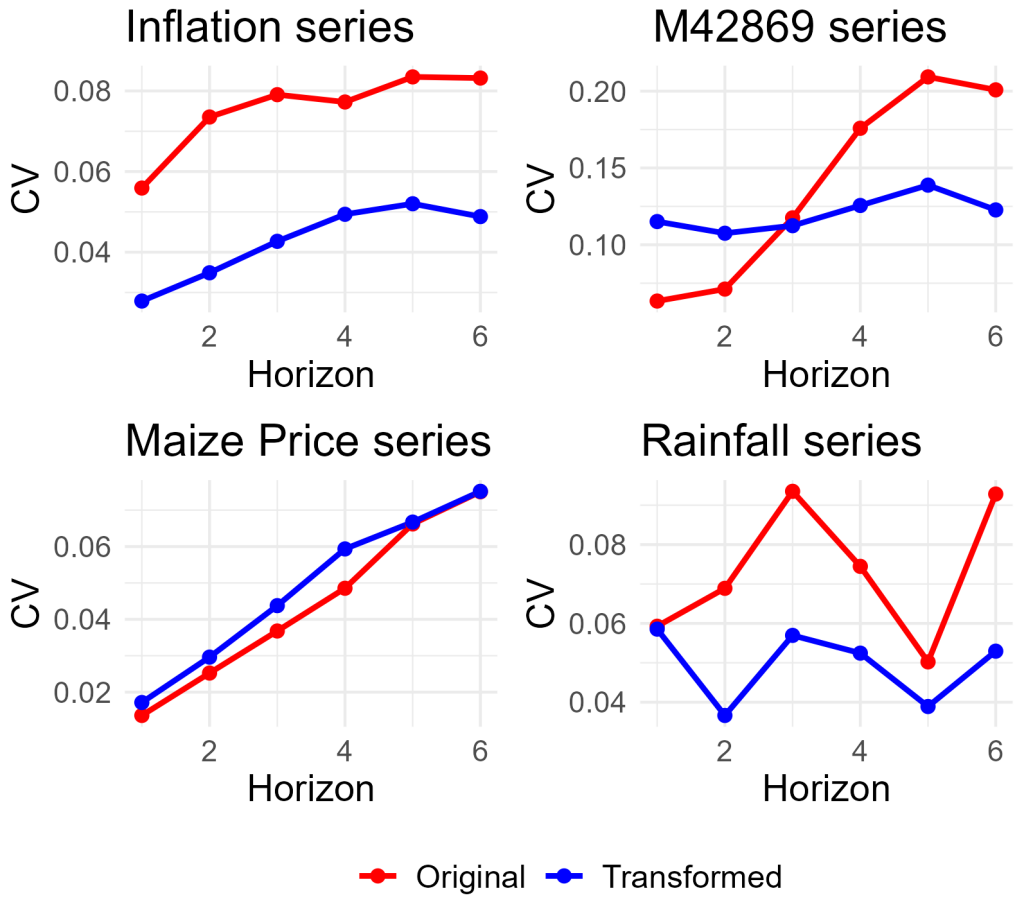


Fig. 1. Coefficient of Variation of forecasts per horizon

MAPE generally increases with the forecasting horizon. Before transformation, M42869 and inflation exhibit the highest errors, while maize prices and rainfall show comparatively lower MAPE values. After transformation, forecast accuracy improves notably for the inflation and M42869 series. The model achieves its lowest MAPE on the transformed inflation series compared to the other three series. In contrast, rainfall and maize prices show only marginal changes in accuracy, with a slight increase in MAPE. In absolute terms, M42869 and maize prices maintain the highest percentage errors after transformation.

A comparison with classical `autoarima` models is presented in Table 2. Before transformation, ARIMA outperforms LSTM on all series except rainfall. After transformation, LSTM outperforms ARIMA on inflation and M42869, where lumpiness has been substantially reduced. Overall, the model performs best

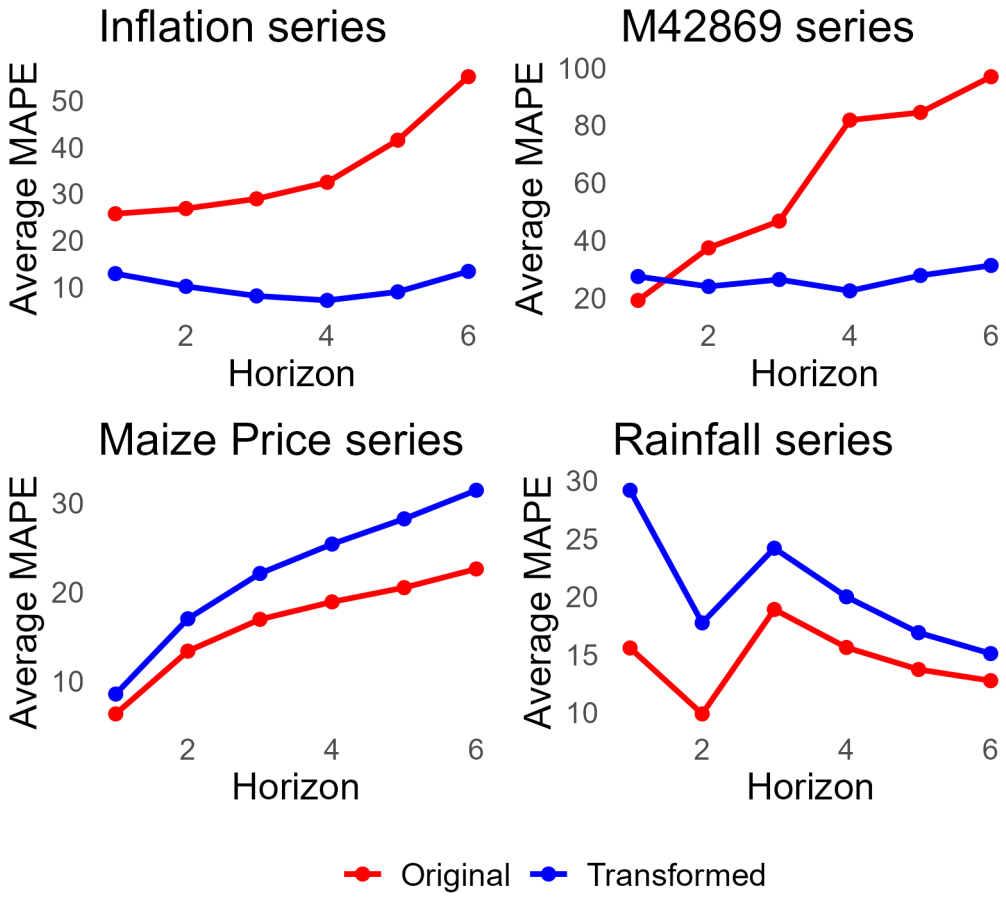


Fig. 2. Mean Absolute Percentage Error across the horizons

on the transformed inflation and rainfall series, both of which exhibit strong homogeneous patterns.

One possible explanation for these results is that, in series with high variability, it is difficult for the model to extract a clear signal from noise. Contrary to the common assertion that the LSTM gating mechanism can effectively handle noise, it may either discard useful information or retain irrelevant fluctuations, leading to instability and reduced accuracy. For smoother series, such as the transformed inflation series, the model can more easily adapt to the underlying signal. This behaviour is consistent with (Greff *et al.* , 2016), who showed that input noise negatively affects the performance of recurrent neural networks.

Series	MAPE (Original)	MAPE (Transformed)	MAPE (ARIMA)
Inflation	35.11	10.13	16.10
Rainfall	14.43	20.53	20.72
Price	16.45	22.41	11.88
M42869	61.10	26.60	39.68

Table 2. Comparison of LSTM and ARIMA MAPE values before and after transformation

The relatively high MAPE observed in transformed M42869 and maize prices is likely due to the presence of dual patterns, namely the coexistence of strong trend and seasonal components. LSTMs generally perform better when a series is dominated by a single strong and homogeneous pattern, as documented by (Hewamalage *et al.* , 2021). It is therefore not surprising that, after transformation, the model performs better on the inflation and rainfall series, which are characterized by more homogeneous patterns.

These findings are also consistent with (Yadav , 2024), who reported that LSTM models were outperformed by ARIMA in forecasting wheat prices, suggesting that LSTMs may struggle with commodity price series that exhibit complex or less predictable structures.

7. Conclusion

This study examined the forecasting performance of Long Short-Term Memory models (LSTMs) on time series with distinct temporal variations. The analysis was conducted using three Ugandan time series (inflation, maize price, and rainfall) alongside the M42869 series from the M4 forecasting competition. The findings, based on evaluations of both model uncertainty and accuracy, can be summarized as follows:

The model performed poorly on the two series with high lumpiness. Forecasts from independent training runs exhibited a high coefficient of variation, indicating high levels of uncertainty. Mean Absolute Percentage Errors were also highest for these series.

When transformations effectively reduced lumpiness, particularly in the inflation and M42869 series, forecast accuracy and uncertainty improved substantially. In contrast, transformations had limited or even slightly negative effects on already stable series, such as rainfall and maize prices.

Although LSTMs handle trend and seasonality well when each component is predominantly homogeneous, their performance declines in the presence of complex, interacting structures. Series with simultaneously strong trend and seasonal components, such as maize prices and M42869, remain challenging even after transformation. In such cases, additional preprocessing, such as

deseasonalization, may be beneficial.

Overall, these results highlight that LSTM forecasting performance is highly sensitive to the structural properties of the input series. Tailoring preprocessing strategies, particularly transformation techniques, to the characteristics of each series is essential for achieving robust and reliable forecasts. These insights can inform the future development and deployment of LSTM models in applied forecasting contexts.

Declarations

Conflict of Interest

The authors declare that there is no conflict of interest.

Funding

No funding was received for conducting this study.

Data Availability

The datasets utilized in this study are publicly available from the following sources:

- Maize price series: Food Price and Monitoring tool of the Food and Agriculture Organisation(FAO) https://fpma.fao.org/giews/fpmat4/#/dashboard/tool/domestic
- Rainfall volumes: Data Viz platform of world Food Programme (WFP) https://dataviz.vam.wfp.org/eastern-africa/uganda/climate-explorer
- Inflation series: Data Viz platform of world Food Programme (WFP)https://dataviz.vam.wfp.org/economic/export-data/inflation
- M4-42869 series: M4 competition series https://www.kaggle.com/datasets/yogesh94/m4-forecasting-competition-dataset

Ethics Approval

Ethical approval was not required for this study as it did not involve human participants or animals.

Acknowledgement

The authors acknowledge the helpful comments and suggestions provided by Prof. Johan Lyhagen, Department of Statistics, Uppsala University. We also acknowledge the College of Business and Management Sciences (CoBAMS), Makerere University, for research support to the corresponding author.

Appendix A: Appendices

Dataset	LSTM Units (Layer 1, Layer 2)	Dropout Rate
Maize price	128, 64	0.2
Transformed Maize price	64, 64	0.3
Inflation	64, 64	0.2
Transformed Inflation	128, 128	0.3
Rainfall	64, 64	0.2
Transformed Rainfall	64, 128	0.2
M42869	128, 64	0.3
Transformed M42869	128, 64	0.3

Table A1. Best LSTM parameters for each dataset

Dataset	**Model**
Maize price	$ARIMA(1,1,0)$
Inflation	$ARIMA(4,1,0)(1,0,1)[12]$
Rainfall	$ARIMA(0,0,0)(2,0,0)[12]$ w/ mean
M42869	$ARIMA(1,0,0)(1,1,0)[12]$

Table A2. Best ARIMA model for each series

References

Mehedi, Md Abdullah Al, Khosravi, Marzieh, Yazdan, Munshi Md Shafwat, and Shabanian, Hanieh, 2022. Exploring temporal dynamics of river discharge using univariate long short-term memory (LSTM) recurrent neural network at East Branch of Delaware River. *Hydrology*, 9 (11), 202.

Hochreiter, Sepp and Schmidhuber, Jürgen, 1997. Long short-term memory. *Neural Computation*, 9 (8), 1735–1780.

Rguibi, Mohamed A., Moussa, Nabil, Madani, Abdelhak, Aaroud, Abdelaziz, and Zine-Dine, Khalid, 2022. Forecasting COVID-19 transmission with ARIMA and LSTM techniques in Morocco. *SN Computer Science*, 3 (2), 133.

Shafi, Imran, Sohail, Asad, Ahmad, Junaid, Espinosa, Juan C. M., López, Luis A. D., Thompson, Eric B., and Ashraf, Imran, 2023. Spare parts forecasting and lumpiness classification using neural network model and its impact on aviation safety. *Applied Sciences*, 13 (9), 5475.

Yadav, Ankit, 2024. A comparative study of time series, machine learning, and deep learning models for forecasting global price of wheat. *Operations Research Forum*, 5 (4), 113.

Kang, Yanfei, Hyndman, Rob J., and Smith-Miles, Kate, 2017. Visualising forecasting algorithm performance using time series instance spaces. *International Journal of Forecasting*, 33 (2), 345–358.

Werbos, Paul J., 1990. Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, 78 (10), 1550–1560.

Cedrez, Carlos B., Chamberlin, Jordan, and Hijmans, Robert J., 2020. Seasonal, annual, and spatial variation in cereal prices in Sub-Saharan Africa. *Global Food Security*, 26, 100438.

Nsubuga, Fred W. N., Botai, Joel O., Olwoch, Joseph M., Rautenbach, Christien J. d. W., Bevis, Y., and Adetunji, Adewale O., 2014. The nature of rainfall in the main drainage sub-basins of Uganda. *Hydrological Sciences Journal*, 59 (2), 278–299.

Kemoe, L., Mbohou, M., Mighri, H., and Quayyum, S. N., 2024. *Effect of Exchange Rate Movements on Inflation in Sub-Saharan Africa*. International Monetary Fund.

World Food Programme, 2024. WFP DataViz Platform — Eastern Africa. https://dataviz.vam.wfp.org.

Makridakis, Spyros, Spiliotis, Evangelos, and Assimakopoulos, Vassilios, 2020. The M4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36 (1), 54–74.

Kumar, Yogesh, 2018. M4 Forecasting Competition Dataset. https://www.kaggle.com/datasets/yogesh94/m4-forecasting-competition-dataset.

Hyndman, Rob J. and Athanasopoulos, George, 2021. *Forecasting: Principles and Practice*. 3rd edition. OTexts, Melbourne, Australia. https://otexts.com/fpp3.

Allaire, J. J., Chollet, François, and RStudio, 2018. tensorflow: R Interface to 'TensorFlow'. R package version 2.1.5. https://tensorflow.rstudio.com.

Bala, R., Singh, R. P., and others, 2019. Financial and Non-Stationary Time Series Forecasting Using LSTM Recurrent Neural Network for Short and Long Horizon. *Proceedings of the 2019 10th International Conference on Computing, Com-*

munication and Networking Technologies (ICCCNT), 1–7. IEEE, Kanpur, India. doi:10.1109/ICCCNT45670.2019.8944624.

Smyl, Slawek, 2020. A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting*, 36 (1), 75–85.

Greff, Klaus, Srivastava, Rupesh K., Koutník, Jan, Steunebrink, Bas R., and Schmidhuber, Jürgen, 2016. LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28 (10), 2222–2232.

Giannopoulos, Panagiotis G, Dasaklis, Thomas K, Tsantilis, John, and Patsakis, Constantinos, 2025. Machine learning algorithms in intermittent and lumpy demand forecasting: A review. *Available at SSRN 5231788*.

Singh, Sanjeev, Mukherjee, Asmita, Panda, Jagabandhu, Choudhury, Animesh, and Bhattacharyya, Saugat, 2024. Analysis and forecasting of temporal rainfall variability over hundred Indian cities using deep learning approaches. *Earth Systems and Environment*, 8 (3), 599–625. Springer Nature BV.

Gers, Felix A, Schmidhuber, Jürgen, and Cummins, Fred, 2000. Learning to forget: Continual prediction with LSTM. *Neural computation*, 12 (10), 2451–2471. MIT press.

Valdenegro-Toro, Matias, de Jong, Ivo Pascal, and Zullich, Marco, 2024. Unified Uncertainties: Combining Input, Data and Model Uncertainty into a Single Formulation. *arXiv preprint arXiv:2406.18787*.

Xu, Lei, Chen, Nengcheng, and Yang, Chao, 2021. Quantifying the uncertainty of precipitation forecasting using probabilistic deep learning. *Hydrology and Earth System Sciences Discussions*, 2021, 1–27. Göttingen, Germany.

Hewamalage, Hansika, Bergmeir, Christoph, and Bandara, Kasun, 2021. Recurrent Neural Networks for Time Series Forecasting: Current status and future directions. *International Journal of Forecasting*, 37 (1), 388–427. Elsevier.

Food and Agriculture Organization of the United Nations, 2026. Food Price and Monitoring Tool: Domestic Maize Prices. https://fpma.fao.org/giews/fpmat4/#/dashboard/tool/domestic. Accessed: 2026-02-03.

World Food Programme, 2026. Data Viz Platform: Rainfall Volumes in Uganda. https://dataviz.vam.wfp.org/eastern-africa/uganda/climate-explorer. Accessed: 2026-02-03.

Makridakis, Spyros and others, 2018. M4 Competition Series: M4-42869. https://www.kaggle.com/datasets/yogesh94/m4-42869. Accessed: 2026-02-03.

Lakshminarayanan, Balaji, Pritzel, Alexander, and Blundell, Charles, 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30.

Hyndman, Rob, Kang, Yanfei, Montero-Manso, Pablo, O'Hara-Wild, Mitchell, Talagala, Thiyanga, Wang, Earo, and Yang, Yangzhuoran, 2026. tsfeatures: Time Series Feature Extraction. R package version 1.1.1.9000. https://pkg.robjhyndman.com/tsfeatures/.