

ENHANCING RAINFALL FORECASTING ACCURACY: A COMPARATIVE STUDY OF ARIMA, ANN, AND HYBRID MODELS USING AGRO-CLIMATIC DATA

MOHAMMED SAMEER, R. K. S. RAJPUT, DIBYANSHU, SALONI THUKRAL,
AND SUMAN BIJLWAN

ABSTRACT. Agro-climatic data from Pantnagar, spanning the years January 2010 to March 2022, is used in the study to assess the forecasting capabilities of Auto Regressive Integrated Moving Average (ARIMA), Artificial Neural Networks (ANN), and hybrid models. These prediction models were trained and tested on the dataset, and Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE) were used to evaluate the models' performance. With the lowest MSE and RMSE values, the Long Short-Term Memory (LSTM) model outperformed the others, demonstrating its ability to accurately capture the temporal relationships in the rainfall data. Additionally, the Temporal Long Short-Term Memory Network (TLNN) demonstrated a solid balance between performance and complexity. Despite being simpler, the Feedforward Neural Network (FNN) model performed less accurately when handling time series data. The hybrid model did not perform better than the separate models, indicating that accuracy may not always be increased by combining ANN predictions with ARIMA residuals.

2000 MATHEMATICS SUBJECT CLASSIFICATION. 62M10, 86A10.

KEYWORDS AND PHRASES. ARIMA, ANN, Hybrid models, Agri-climatic data.

SUBMISSION DATE: 03.08.2024

1. INTRODUCTION

The primary aim of this paper is to enhance rainfall forecasting accuracy by comparing ARIMA, ANN, and hybrid models using agri-climatic data. This study contributes by introducing a comprehensive evaluation of these models, focusing on their ability to capture linear and non-linear patterns in rainfall data. Through rigorous experimentation and performance assessment using RMSE, MSE, and MAE metrics, the findings provide valuable insights into model effectiveness. While most studies focus on either traditional statistical models or machine learning techniques, this paper bridges the gap by investigating hybrid approaches especially the combination of statistical and ANN models, ensuring a more holistic understanding of rainfall forecasting. The results reinforce the importance of integrating both linear and non-linear components for improved prediction accuracy, fostering further research and practical applications in agricultural planning and water resource management.

Since rain supplies most of the water required to sustain crop production, it is crucial to the Indian agricultural system. In India's predominantly rain-fed agriculture, enough and appropriate rainfall is essential for crop success. Accurate rainfall forecasting is essential for agricultural operations, water resource management, and disaster preparedness.

A collection of a variable's observations arranged chronologically throughout time is called a time series. In order to create models that try to replicate the basic process that creates data, time series forecasting makes use of historical observations of the target variable. These models may then be used to make forecasts for the future. A lot of work has gone into creating precise time series forecasting models (Zhang, 2003).

ARIMA (Autoregressive Integrated Moving Average) models are among the most widely used traditional time series models. Their popularity is mostly due to their statistical properties and the popular Box-Jenkins model-building techniques. ARIMA models can handle autoregressive (AR), moving average (MA), and non-stationarity components. Their presumption of linear correlations between previous data (in the AR) or errors (in the MA) is a major disadvantage, though. As a result, non-linearity in the data is not captured by ARIMA models. Additionally, when simulating complicated real-world systems that exhibit non-linear behavior, it is sometimes impractical to meet the assumptions required by ARIMA models prior to modeling.

Conversely, non-linearity in the data may be captured using Artificial Neural Networks (ANNs). Since they don't need pre-specifying the model's structure, they are a data-driven method suitable for situations where the underlying data-generating process is ambiguous and complicated.

The challenge of identifying whether an observed time series is the result of a linear or non-linear data-generating process is the idea behind creating hybrid models. As a result, forecasters have difficulty selecting the right model for their particular circumstance. The optimal model is chosen based on performance metrics like accuracy after several models are constructed using the data at hand. This approach is not perfect, though, due to sample variability, model uncertainty, and potential structural alterations in the data.

Furthermore, a real-world process is probably a combination of both linear and non-linear, therefore it is impractical to assume that it is either one or the other. As a result, combining models that manage linearity and non-linearity to create hybrid models that can handle both features frequently yields superior results. Results from a number of research on ARIMA and ANN have been inconsistent (Al-Saba et al., 1999; Elkatib et al., 1998; Patil, 1990; Tang & Fishwick, 1993; Tang et al., 1991; Zhang et al., 2001). However, a number of forecasting studies have discovered that integrated models outperform solo models in terms of accuracy.

Since the famous M contests (Makridakis et al. 1982), it has been observed that combinations are frequently included in the most accurate predicting models. The accuracy of these models decreases as the prognosis advances. Clemen (1989) provided a comprehensive overview of the topic, whereas Reid (1968) and Bates & Granger (1969) carried out the first studies in this field. Combining several models makes sense since each one is very good at seeing unique patterns in the data. Numerous studies have shown how good these integrated techniques are at predicting (Makridakis, 1989; Newbold & Granger, 1974; Palm & Zellner, 1992; Winkler, 1989). This is further supported by recent work by Mathpal et al. (2023), which highlights the importance of timely and enough rainfall for crop development and shows the increased accuracy and reliability of ensemble machine learning models for rainfall forecasting in Indian agriculture. Furthermore, developments in agricultural automation, like the creation of effective CNN-based image classification models for

the detection of cabbage weeds (Dibyanshu and Rajput 2024), demonstrate the continuous advancements in machine learning models to boost agricultural productivity and efficiency.

A number of combinations have been proposed for neural network forecasting. Wedding and Cios (1996) have addressed the integration of radial basis function networks with Box-Jenkins models. Luxhoj et al. (1996) blended artificial neural networks (ANN) with econometric methodologies for sales forecasting. Additionally, Ginzburg & Horn (1993) used several feedforward neural networks to increase prediction accuracy.

Although a large body of research has shown that hybrid models often perform better than individual models, some studies have shown that individual models can produce outcomes that are on par with or even better than those of hybrid models. Makridakis et al. (1982) noted that basic exponential smoothing frequently outperformed more sophisticated techniques, such as hybrid models, during the first M contests. Individual models, such as autoregressive (AR) models, performed exceptionally well in macroeconomic forecasting, according to Stock and Watson (1999). Armstrong (2001) observed that in terms of accuracy and usability, simpler, well-defined individual models often outperformed combination models. In several datasets, Hyndman & Khandakar (2008) discovered that ARIMA models performed better than hybrid models. In a similar vein, Zhang (2003) also found cases in which certain datasets or scenarios demonstrated the superiority of distinct models. Using the R program, Pant A. & Rajput R.S. (2019) detailed the building of time series models for predicting the price of gold, with the objective of analyzing historical price trends and improving forecasting accuracy for financial decision-making.

The paper is organized as follows. In Section 1, we introduce the study and its objectives. Section 2 presents the methodology, detailing the development of statistical, ANN, and hybrid models. Section 3 discusses the results and provides an in-depth analysis. Finally, Section 4 concludes the study and outlines potential directions for future research.

2. METHODOLOGY

Study Area: The study's primary emphasis was Pantnagar, which is situated in Uttarakhand, India's Kumaon area. Geographically, Pantnagar is located in latitude 29.021038N and longitude 79.489738E, encompassing an area of around 51 square kilometers. The South West monsoon season, which runs from July to September, is when the region receives the majority of its rainfall. In Pantnagar, winter temperatures may drop as low as 8°C and rise as high as 15°C, while summer temperatures can range as low as 26°C and as high as 46°C.

Data Source: We used daily agri-climatic data from the NASA POWER database for our study in Pantnagar, Uttarakhand, India. The National Aeronautics and Space Administration (NASA) is a pioneer in Earth Science research, supporting satellite systems and conducting studies that provide critical data for understanding climate and climatic processes. NASA's Earth Science research program has made important datasets that are necessary for researching and evaluating different climatic aspects accessible.

Python: Python is now widely used in software development as a flexible programming language. Because of its widely recognized readability, ease of use, and

extensive library support, it is a popular choice for developing a wide range of applications. Python is an open-source, high-level language known for its elegant syntax, which emphasizes code readability and simplicity. This characteristic enables developers to convey ideas and concepts using fewer lines of code compared to other languages, catering to both beginners and seasoned programmers alike. Additionally, Python boasts various specialized libraries such as NumPy for multidimensional arrays and numerical computations, OpenCV (cv2) for image manipulation, and Matplotlib for creating diverse plots and visualizations. Google Colab is employed for coding, debugging, and execution.

Autoregressive Integrated Moving Average (ARIMA) model: The future value of a variable is shown as a linear function of several past observations and random errors in an autoregressive integrated moving average (ARIMA) model. The underlying process that generates the time series is expressed as follows (Box et al., 2015):

$$(1) \quad y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \theta_0 \epsilon_t + \theta_1 \epsilon_{t-1} + \cdots + \theta_q \epsilon_{t-q}$$

The actual value and random error at time-period t are denoted by y_t and ϵ_t , respectively. The model parameters are the coefficients ϕ_i (for $i = 1, 2, \dots, p$) and θ_j (for $j = 0, 1, 2, \dots, q$), where p and q are integers that are referred to as the model's orders. With a mean of zero and a fixed variance σ^2 , it is assumed that the random errors ϵ_t are independently and identically distributed. This equation contains several important special instances in the ARIMA family of models. It reduces to a moving average (MA) model of order q if $p = 0$, and to an autoregressive (AR) model of order p if $q = 0$. One of the most important steps in creating an ARIMA model is determining the appropriate model order (p, q) .

Box and Jenkins's useful approach to creating ARIMA models, which was developed on Yule's 1926 work, has immensely improved time series analysis and forecasting applications. The Box-Jenkins approach consists of three iterative steps: diagnostic checking, parameter estimation, and model identification.

The fundamental idea behind model identification is that a time series should have specific theoretical autocorrelation properties if it is the outcome of an ARIMA process. One or more viable models for the given time series are frequently revealed by comparing the theoretical and real autocorrelation patterns. The autocorrelation function (ACF) and partial autocorrelation function (PACF) of the sample data are the main tools for this aim, according to Box and Jenkins.

To develop an appropriate ARIMA model for forecasting, stationarity must typically be achieved in the identification stage, which calls for data processing. The mean and autocorrelation structure of a stationary time series remain constant across time. Before fitting an ARIMA model, differencing and power transformations are frequently used to remove trends and stabilize variance in time series that show heteroscedasticity and trends.

Once a rough model is established, estimating the model parameters is simple. Through parameter estimation, an overall measure of errors may be minimized via a nonlinear optimization procedure.

Artificial Neural Network (ANN): An efficient method for solving a variety of nonlinear issues is the use of artificial neural networks, or ANNs. The capacity of ANNs to properly represent a broad range of functions makes them valuable as

universal approximators, which is one of their primary advantages. This adaptability stems from their capacity to handle data in parallel. In contrast to conventional models, an ANN does not rely on preset model assumptions; instead, its network topology is dictated by the data itself.

Three layers of processing units connected by acyclic linkages make up a single hidden layer, a common feedforward network for time series forecasting. The output y_t relates to the inputs $(y_{t-1}, y_{t-2}, \dots, y_{t-p})$ through the equation.

$$(2) \quad y_t = \alpha_0 + \sum_{j=1}^q \alpha_j g \left(\beta_{0j} + \sum_{i=1}^p \beta_{ij} y_{t-i} \right) + \epsilon_t$$

In equation 2, α_j and β_{ij} are connection weights, with p input nodes and q hidden nodes. The logistic function, $g(x) = \frac{1}{1+\exp(-x)}$, is typically used as the hidden layer transfer function. This ANN model achieves nonlinear mapping from past values to the future value (y_t):

$$(3) \quad y_t = f(y_{t-1}, y_{t-2}, \dots, y_{t-p}; W) + \epsilon_t$$

Even though its structure is simple, this model works as a nonlinear autoregressive model, often with a single output node for one-step-ahead forecasting, and is quite successful. Where W comprises all parameters and f is specified by the network topology and weights.

When the network structure (p, q) is given, the network is trained to estimate parameters. Parameter optimization aims to lower the mean squared error, just like ARIMA models do. The generalized reduced gradient (GRG2) and other advanced nonlinear optimization techniques that go beyond basic backpropagation are used to achieve this.

Hybrid Method: We can decompose time series data into two components: a linear trend (L_t) and a non-linear component (N_t), expressed as

$$(4) \quad y_t = L_t + N_t$$

Initially, the ARIMA model is employed to capture the linear trend. The residuals

$$(5) \quad e_t = y_t - \hat{L}_t$$

Which represent the non-linear component not accounted for by ARIMA, can then be used for further modeling. An artificial neural network (ANN) is applied to these residuals, with n input nodes, to model the non-linear dynamics:

$$(6) \quad e_t = f(e_{t-1}, e_{t-2}, \dots, e_{t-n}) + \epsilon_t$$

Where f denotes a neural network-derived non-linear function and ϵ_t represents random error. The combined forecast is derived as

$$(7) \quad \hat{y}_t = \hat{L}_t + \hat{N}_t$$

reflecting a hybrid approach involving two main steps: (1) modeling the linear component using ARIMA, and (2) modeling the residual non-linear component using

ANN. This method enables separate modeling and integration of linear and non-linear components in time series forecasting.

All models were trained and tested on monthly rainfall data of Pantnagar, obtained from NASA POWER, Monthly data on rainfall was split into approximately 79% training data and 21% testing data. The optimal ARIMA model was selected based on the minimum AIC value, while the parameters for ANN models (FNN, TLNN, LSTM) were determined using the grid search method. Further details on model performance are discussed in the Results section .

RESULTS AND DISCUSSION

ARIMA Model: The ARIMA model was applied to monthly rainfall data of Pantnagar for the period January 2010 to March 2022 (Fig.1)

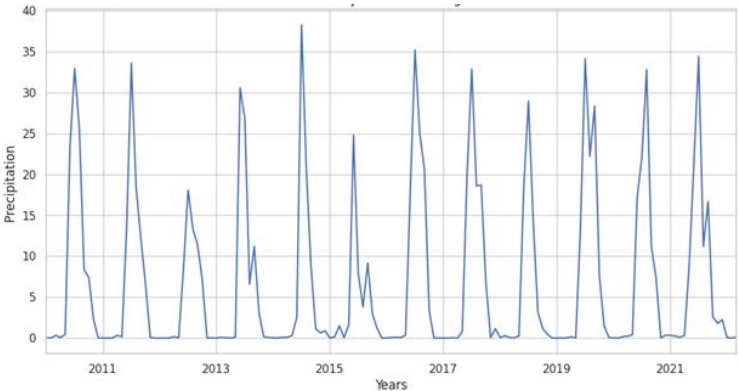


FIGURE 1. Monthly rainfall data of Pantnagar from January 2010 to March 2022

The dataset was divided into a training set (before the previous 30 months) and a test set (the last 30 months) (Fig 3). Stepwise search is used to minimize the Akaike Information Criterion (AIC), and the best-fitting ARIMA model was found to be ARIMA (0, 0, 1) (2, 0, 0) [12] with an intercept term. Table 3 lists a few of the fitted ARIMA models along with the accompanying AIC values.

TABLE 1. ARIMA Model Performance Metrics

Metric	Value
Log Likelihood	-374.19
AIC	758.38
Ljung-Box test (L1) p-value	0.77
Jarque-Bera test p-value	0.001
Mean Squared Error (MSE)	21.29
Root Mean Squared Error (RMSE)	4.61

With an AIC of 758.38 and a log likelihood of -374.19, the ARIMA (0,0,1) (2,0,0)[12] model fits the data rather well. The Ljung-Box test (L1) p-value of 0.77 suggests that there is no significant autocorrelation in the residuals, while the

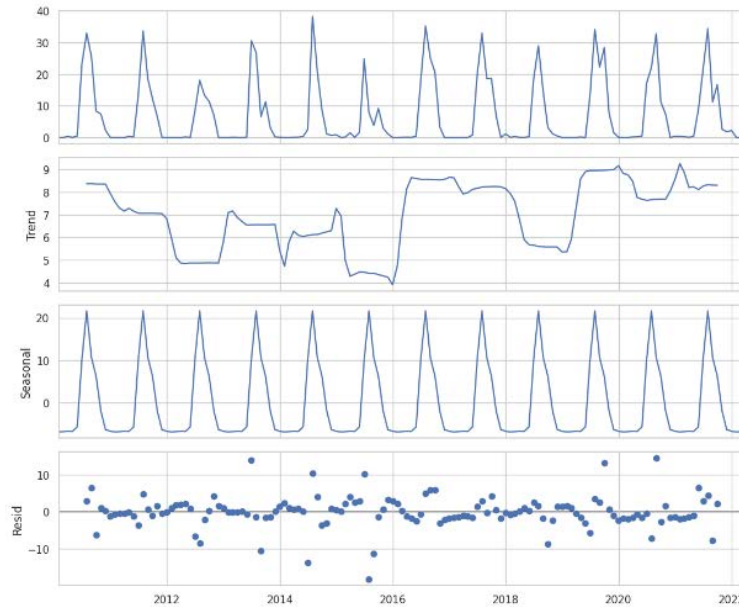


FIGURE 2. Decomposition of original series into Trend and Seasonal components

TABLE 2. **ARIMA Model Parameters**

Parameter	Value
Intercept	1.079
Moving Average (MA) coefficient (Lag 1)	0.37
Seasonal Autoregressive (AR) coefficients (Lags 12 and 24)	0.25, 0.59
Residual variance (σ^2)	30.31

Jarque-Bera test p-value of less than 0.001 indicates that the residuals are not normal. With a Mean Squared Error (MSE) of 21.29 and a Root Mean Squared Error (RMSE) of 4.61, the model's performance metrics demonstrate its predictive accuracy. A prediction for the upcoming three years was generated using this model.

Artificial Neural Network Models Three ANN models- Long Short-Term Memory (LSTM), Time-Lagged Neural Network (TLNN), and Feedforward Neural Network (FNN) were trained, implemented, and evaluated using Python on the same data. Each model was tested with various parameter combinations (Table 4) and the best-performing configuration was selected (Table 5) based on performance measure

Fig 5-7 depict the predicted and actual values for the test set corresponding to each ANN model. The LSTM model demonstrated superior performance compared to the others based on performance measures.

Hybrid Model: ARIMA residuals and ANN predictions were combined to form a hybrid model. The best ANN model was trained using the residuals from the ARIMA model on the training set, and it forecasted residuals for the next 30 months.

TABLE 3. ARIMA Models with AIC, MAE, MSE, RMSE, MAPE values

ARIMA Model	AIC	MAE	MSE	RMSE	MAPE
ARIMA(0,0,0)(0,0,0)[12]	885.00	7.85	96.07	9.80	42656.1
ARIMA(1,0,0)(1,0,0)[12]	802.54	5.65	45.99	6.78	34216.21
ARIMA(0,0,1)(0,0,1)[12]	820.84	7.52	83.03	9.11	41897.48
ARIMA(1,0,0)(0,0,1)[12]	833.86	8.95	112.9	10.6	46589.03
ARIMA(0,0,0)(1,0,0)[12]	813.41	4.40	30.78	5.54	24723.62
ARIMA(2,0,0)(1,0,0)[12]	802.98	5.35	44.26	6.65	33216.14
ARIMA(1,0,1)(1,0,0)[12]	803.17	5.47	44.96	6.70	33649.6
ARIMA(0,0,1)(1,0,0)[12]	801.61	5.15	41.15	6.41	32209.49
ARIMA(0,0,1)(0,0,0)[12]	821.91	8.09	100.0	10.0	43185.22
ARIMA(0,0,2)(0,0,2)[12]	758.38	3.21	21.09	4.61	12114.53
ARIMA(1,0,1)(0,2,0)[12]	759.64	3.20	22.02	4.69	12871.12
ARIMA(0,0,2)(2,0,0)[12]	759.67	3.18	21.81	4.67	12726.49
ARIMA(0,0,1)(2,0,0)[12]	762.18	2.71	19.50	4.41	8995.185

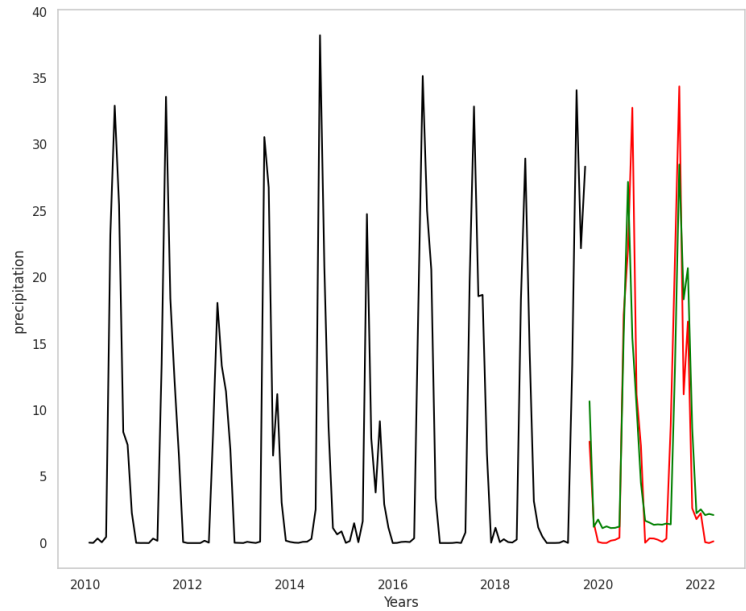


FIGURE 3. Train set(black),Test set(red), Test forecast(green)

TABLE 4. Hybrid Model Performance Metrics

Metric	Value
Mean Squared Error (MSE)	21.957
Root Mean Squared Error (RMSE)	4.686
Mean Absolute Error (MAE)	2.692

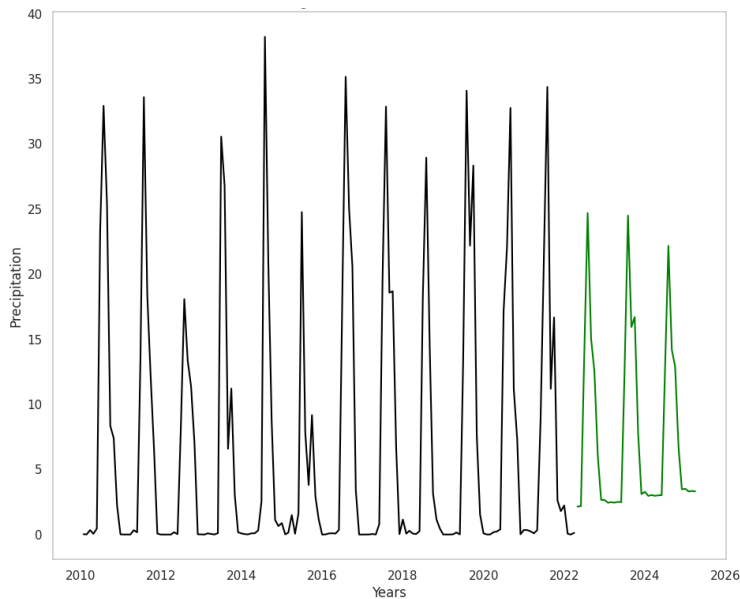


FIGURE 4. ARIMA model forecast for 3 years (2023 to 2025)

TABLE 5. Model Parameters and Performance measures for selected models

Model	Parameters	MSE	RMSE	MAE
LSTM	Look-back: 12, Hidden nodes: 5, Epochs: 50, Batch size: 1, Future steps: 30	13.877	3.725	2.678
FNN	Look-back: 36, Hidden nodes: 5, Output nodes: 1, Epochs: 50, Batch size: 1, Future steps: 30	19.981	4.470	2.801
TLNN	Time-lagged points: [1, 2, 3, 12, 24, 36], Hidden nodes: 5, Output nodes: 1, Epochs: 50, Batch size: 1, Future steps: 30	17.375	4.168	2.479

These predicted residuals were added to the ARIMA forecasted values to obtain the final forecast.

Comparison of Models: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) were used to compare the performance of the ARIMA, ANN, and hybrid models.

Incorporating advanced rainfall forecasting models, holds significant potential for enhancing agricultural practices. Accurate rainfall predictions enable farmers to optimize irrigation scheduling, ensuring efficient water use and potentially reducing costs. Moreover, precise weather forecasts assist in the timely application of fertilizers and pesticides. Applying these inputs under favorable weather conditions enhances their effectiveness and minimizes environmental impact. Additionally, accurate rainfall forecasting aids in planning fieldwork activities.

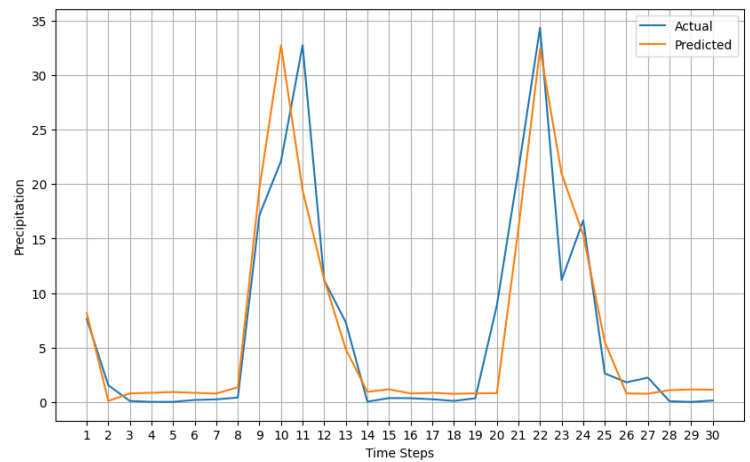


FIGURE 5. TLNN Model: Actual vs Predicted

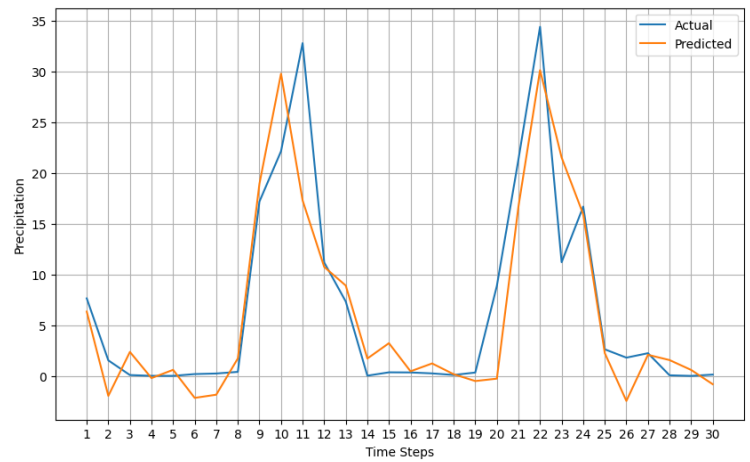


FIGURE 6. FNN Model: Actual vs Predicted

TABLE 6. Summary of Performance measures of all models

Model	MSE	RMSE	MAE
ARIMA	21.291	4.614	3.033
LSTM	13.877	3.725	2.678
FNN	19.981	4.470	2.801
TLNN	17.375	4.168	2.479
Hybrid	21.957	4.686	2.692

By leveraging such predictive models, agricultural stakeholders can make informed decisions that enhance productivity, resource management, and sustainability. The integration of advanced forecasting tools into agricultural practices represents a

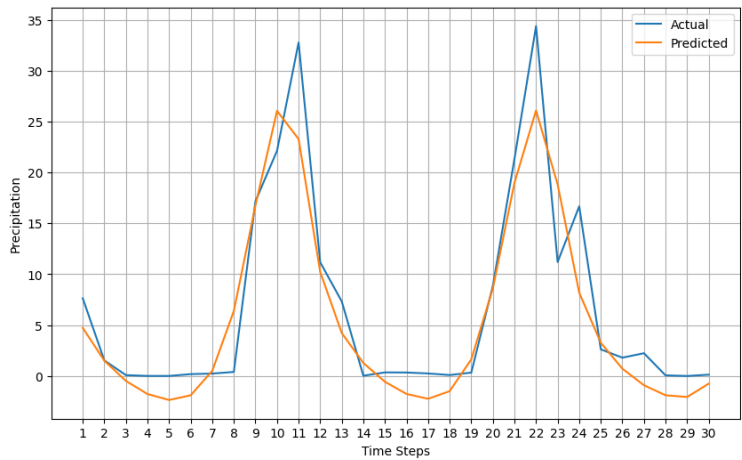


FIGURE 7. LSTM Model: Actual vs Predicted

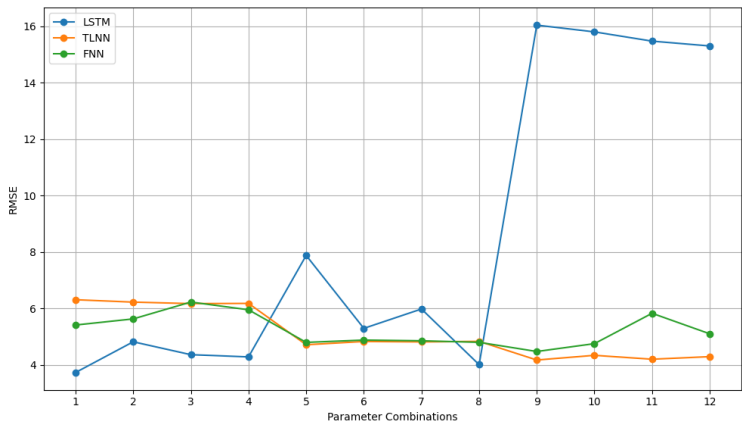


FIGURE 8. RMSE score comparison for LSTM, TLNN AND FNN for different parameter combinations

proactive approach to managing weather-related risks and optimizing farm operations.

Conclusion: The LSTM model demonstrated superior performance, achieving the lowest MSE and RMSE, making it the most suitable for rainfall prediction in Pantnagar. TLNN also showed promising results, balancing accuracy and model complexity. The ARIMA model, though outperformed by ANN models, remains valuable for understanding underlying trends. The hybrid approach requires further refinement to enhance predictive performance.

These findings have significant implications for agricultural planning and water resource management. Future research should explore the integration of climate indices, soil moisture, and remote sensing data to further enhance model accuracy.

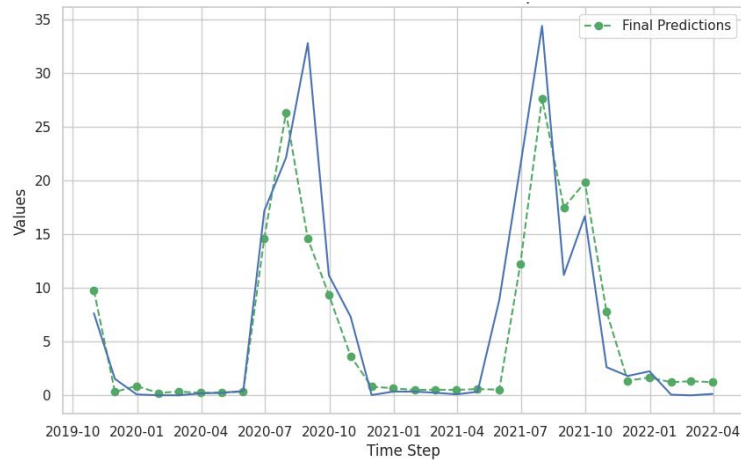


FIGURE 9. Hybrid Model Actual vs Predicted

Additionally, these results can inform policy decisions on sustainable water management and early warning systems, contributing to more resilient agricultural practices

REFERENCES

- [1] Al-Saba, T., & El-Amin, I. (1999). Artificial neural networks as applied to long-term demand forecasting. **Artificial Intelligence in Engineering**, 13(2), 189-197.
- [2] Armstrong, J. S. (2001). **Combining forecasts** (pp. 417-439). Springer US.
- [3] Bates, J. M., & Granger, C. W. (1969). The combination of forecasts. **Journal of the Operational Research Society**, 20(4), 451-468.
- [4] Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). **Time series analysis: Forecasting and control**. John Wiley & Sons.
- [5] Clemen, R. T. (1989). Combining forecasts: A review and annotated bibliography. **International Journal of Forecasting**, 5(4), 559-583.
- [6] Elkateb, M. M., Solaiman, K., & Al-Turki, Y. (1998). A comparative study of medium-weather-dependent load forecasting using enhanced artificial/fuzzy neural network and statistical techniques. **Neurocomputing**, 23(1-3), 3-13.
- [7] Ginzburg, I., & Horn, D. (1993). Combined neural networks for time series analysis. **Advances in Neural Information Processing Systems**, 6.
- [8] Hyndman, R. J., & Khandakar, Y. (2008). Automatic time series forecasting: the forecast package for R. **Journal of Statistical Software**, 27, 1-22.
- [9] Makridakis, S. (1989). Why combining works? **International Journal of Forecasting**, 5(4), 601-603.
- [10] Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., & Winkler, R. (1982). The accuracy of extrapolation (time series) methods: Results of a forecasting competition. **Journal of Forecasting**, 1(2), 111-153.
- [11] Newbold, P., & Granger, C. W. (1974). Experience with forecasting univariate time series and the combination of forecasts. **Journal of the Royal Statistical Society: Series A (General)**, 137(2), 131-146.
- [12] Palm, F. C., & Zellner, A. (1992). To combine or not to combine? Issues of combining forecasts. **Journal of Forecasting**, 11(8), 687-701.
- [13] Pant, A., & Rajput, R. S. (2019). Time Series Analysis of Gold Price Using R. In: **Digitalization in Commerce & Its Impact on Economy**. Social Research Foundation, Vol (1), p. 15-24.

- [14] Patil, R. B. (1990). *Neural networks as forecasting experts: Test of dynamic modeling over time series data* (Doctoral dissertation, Oklahoma State University).
- [15] Reid, D. J. (1968). Combining three estimates of gross domestic product. *Economica*, 35(140), 431-444.
- [16] Tang, Z., & Fishwick, P. A. (1993). Feedforward neural nets as models for time series forecasting. *ORSA Journal on Computing*, 5(4), 374-385.
- [17] Tang, Z., De Almeida, C., & Fishwick, P. A. (1991). Time series forecasting using neural networks vs. Box-Jenkins methodology. *Simulation*, 57(5), 303-310.
- [18] Wedding II, D. K., & Cios, K. J. (1996). Time series forecasting by combining RBF networks, certainty factors, and the Box-Jenkins model. *Neurocomputing*, 10(2), 149-168.
- [19] Winkler, R. L. (1989). Combining forecasts: A philosophical basis and some current issues. *International Journal of Forecasting*, 5(4), 605-609.
- [20] Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, 50, 159-175.
- [21] Zhang, G. P., Patuwo, B. E., & Hu, M. Y. (2001). A simulation study of artificial neural networks for nonlinear time-series forecasting. *Computers & Operations Research*, 28(4), 381-396.

DEPARTMENT OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE, G. B. PANT UNIVERSITY OF AGRICULTURE AND TECHNOLOGY, PANTNAGAR- 263145, UTTARAKHAND, INDIA.

Email address: mohammedsameer120@gmail.com

DEPARTMENT OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE, G. B. PANT UNIVERSITY OF AGRICULTURE AND TECHNOLOGY, PANTNAGAR- 263145, UTTARAKHAND, INDIA.

Email address: rksrajput.bpm@gbpuat.ac.in

DEPARTMENT OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE, G. B. PANT UNIVERSITY OF AGRICULTURE AND TECHNOLOGY, PANTNAGAR- 263145, UTTARAKHAND, INDIA.

Email address: dibyanshubhatt2000@gmail.com

DEPARTMENT OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE, G. B. PANT UNIVERSITY OF AGRICULTURE AND TECHNOLOGY, PANTNAGAR- 263145, UTTARAKHAND, INDIA.

Email address: salonithukral1999@gmail.com

DEPARTMENT OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE, G. B. PANT UNIVERSITY OF AGRICULTURE AND TECHNOLOGY, PANTNAGAR- 263145, UTTARAKHAND, INDIA.

Email address: sumanbijlwan234@gmail.com