JCARM
Journal of Contemporary Academic Research and Methodologies

An Online Peer Reviewed / Refereed Journal
Volume 1 | Issue 4 | May 2026
Website: www.journal.jcarm.in

Research

EVALUATION OF AUTOMATED MODIFIED PROPHET METHOD IN TIME SERIES COMPONENTS IDENTIFICATION

Usman Suleiman[1], Dolapo Shobanke[1], Ajare Emmanuel Oloruntoba[1]*, Shehu Abdullahi[1]

[1]Department of Statistics, Federal University Lokoja, Nigeria.

Correspondence should be addressed to: ajareoloruntoba@gmail.com

Abstract: Time series forecasting is an important analytical tool used in economics, business, environmental studies, and many other fields where understanding temporal patterns is essential for effective planning and decision-making. Among modern forecasting techniques, the Prophet model has gained considerable attention due to its flexibility and ability to model trend and seasonal components in time series data. This study applied Modified Prophet Method (MPM) designed to improve the structural identification capability of the traditional Prophet model. The modification extends the Prophet decomposition framework by introducing an explicit cyclical component alongside the trend, seasonal, and irregular components. The performance of the Modified Prophet Method was compared with the Prophet-style model, Autoregressive Integrated Moving Average (ARIMA), and Exponential Smoothing (ETS). Model performance was assessed using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and residual variance. The results show that the Modified Prophet Method produced the lowest forecasting errors with RMSE of 0.5259, MAE of 0.4221, and residual variance of 0.2789, outperforming the competing models.

Keywords: Stochastic Process, Autocorrelation, Fluctuations, Oscillations.

1. Introduction

Time series analysis is an important statistical tool used to analyze data that are observed sequentially over time. It has been widely applied in fields such as economics, finance, engineering, environmental science, epidemiology, and business forecasting. The main objective of time series analysis is to understand the underlying structure of temporal

 https://doi.org/10.5281/zenodo.20605075

data and to generate reliable forecasts for future observations (Box, Jenkins, Reinsel, & Ljung, 2015).

A time series typically contains several structural components including trend, seasonality, cyclical movements, and irregular fluctuations. The trend component represents the long-term direction of the data, while seasonal variations represent periodic patterns that repeat at regular intervals. Cyclical components capture medium- or long-term oscillations that do not follow fixed seasonal frequencies, whereas irregular components represent random variations caused by unpredictable events (Chatfield, 2004).

One notable advancement in time series forecasting is the Prophet model, developed by Taylor and Letham (2018). Prophet was designed as a scalable and automated forecasting framework capable of handling large volumes of time series data with minimal manual intervention. The model decomposes a time series into interpretable components consisting of trend, seasonality, and holiday effects, while accounting for irregular variations.

In many real-world time series, cyclical fluctuations represent an important structural feature, particularly in economic and environmental data. Failure to explicitly identify such cycles may reduce the interpretability of the model and limit its forecasting performance. Consequently, there is a need to extend the Prophet framework to allow clearer identification of structural components within time series data.

This study therefore proposes a Modified Prophet Method (MPM) designed to improve the structural identification capability of the Prophet model. The proposed approach extends the Prophet framework by incorporating an explicit cyclical component into the decomposition process. By doing so, the Modified Prophet Method aims to enhance the interpretability and analytical usefulness of time series decomposition.

2. Literature Review

A time series can be defined as a sequence of observations recorded in chronological order, typically at equally spaced time intervals such as hourly, daily, monthly, or yearly observations (Wei, 2006). The distinguishing characteristic of a time series data is the presence of temporal dependence, meaning that the value observed at one point in time may be influenced by past observations (Hamilton, 1994).

Time series analysis seeks to model this dependence structure in order to understand the dynamics of the underlying system and produce reliable forecasts. According to Box et al. (2015), the modeling of time series involves identifying systematic patterns within the data while separating these patterns from random noise.

Analyzing time series data generally serves two major objectives of understanding the underlying structure of the data-generating process and forecasting future observations based on historical patterns.

Montgomery, Jennings, & Kulahci, (2015) opined that understanding temporal patterns allows researchers to identify structural features such as growth trends, seasonal variations, and cyclical movements. These components provide insights into the factors influencing the evolution of the time series.

The additive representation of a time series model suggests that observed series results from combined effects of long-term movement, periodic fluctuations, cyclical variations, and random disturbances (Chatfield & Xing, 2019). Interactions among components forms the basis for constructing forecasting models capable of accurately predicting future observations.

2.1 Classical Time Series Decomposition

Classical time series decomposition techniques aim to separate observed data into its constituent components using deterministic procedures. These techniques were among the earliest methods developed for analyzing temporal data (Chatfield, 2004).

The two primary forms of decomposition are commonly used:

2.1.1 Additive Decomposition

$$Y_t = T_t + S_t + C_t + I_t$$

This structure assumes that the magnitude of seasonal fluctuations remains constant regardless of the level of the series.

2.1.2 Multiplicative Decomposition

$$Y_t = T_t \times S_t \times C_t \times I_t$$

This structure assumes that seasonal fluctuations increase proportionally with the level of the time series (Makridakis et al., 1998).

Classical decomposition methods typically involve the following steps:

i. Estimation of the trend using moving averages
ii. Removal of the trend component
iii. Estimation of seasonal indices
iv. Calculation of irregular components

Although classical decomposition provides valuable insights, it has several limitations. The method requires manual intervention and may not perform well when the series contains missing observations or irregular patterns (Chatfield & Xing, 2019).

2.2 ARIMA Models

The Autoregressive Integrated Moving Average (ARIMA) model represents one of the most influential methodologies in time series forecasting. Developed by Box and Jenkins, the ARIMA framework models temporal dependence using autoregressive and moving average processes (Box et al., 2015).

ARIMA models are widely used in economics, finance, and environmental modeling due to their ability to capture autocorrelation structures in stationary time series (Enders, 2015). However, the method assumes linear relationships and may perform poorly when the data contain nonlinear patterns or complex seasonal structures (Makridakis et al., 1998).

2.3 Exponential Smoothing and ETS Models

Exponential smoothing methods represent another important class of forecasting models. These methods generate forecasts using weighted averages of past observations, with more recent observations receiving greater weight (Gardner R.C, 1985).

The ETS framework proposed by Hyndman, Koehler, Ord, and Snyder (2008) represents exponential smoothing models in a state space form, allowing systematic modeling of error, trend, and seasonal components.

The ETS framework classifies models based on three components:

i. Error (additive or multiplicative)
ii. Trend (none, additive, or damped)
iii. Seasonality (none, additive, or multiplicative)

ETS models have demonstrated strong performance in international forecasting competitions such as the M3 competition (Makridakis & Hibon, 2000).

2.4 Prophet Forecasting Model

The Prophet forecasting model was developed by Taylor and Letham (2018) to provide an automated and scalable framework for time series forecasting. The model was designed for business applications where analysts require interpretable models capable of handling large volumes of time series data.

The model incorporates piecewise linear or logistic growth curves to capture nonlinear trends and automatically detects change points in the time series (Taylor & Letham, 2018).

One of key advantage of the Prophet model is its ability to handle missing data, outliers, and irregular observations. These features make it particularly suitable for large-scale business forecasting applications.

However, Prophet primarily focuses on trend and seasonal components. Cyclical patterns that extend beyond seasonal frequencies may be absorbed into the error term, potentially limiting the interpretability of the model.

2.5 Machine Learning Approaches to Time Series Forecasting

The rapid growth of computational power has led to increased interest in machine learning methods for time series forecasting. These methods are capable of modeling complex nonlinear relationships and high-dimensional datasets (Goodfellow, Bengio, & Courville, 2016).

Artificial neural networks (ANNs) have been widely applied to forecasting problems due to their ability to approximate nonlinear functions (Zhang, Patuwo, & Hu, 1998). Recurrent neural networks (RNNs) are particularly suitable for time series data because they incorporate feedback connections that capture temporal dependencies (Hochreiter & Schmidhuber, 1997).

Long short-term memory (LSTM) networks extend the capabilities of RNNs by introducing memory cells that allow the model to capture long-range dependencies in sequential data.

Despite their predictive capabilities, machine learning models often suffer from limited interpretability. Understanding the contribution of individual components to the final forecast can be challenging, making these models less suitable for applications requiring explainability (Shmueli & Lichtendahl, 2016).

3. Methodology

To objectively assess the effectiveness of the Modified Prophet Method (MPM), a structured comparative benchmarking framework is employed. The selected benchmark models represent distinct methodological paradigms in time series analysis: classical stochastic modeling, deep learning–based nonlinear modeling, and automated cloud-based forecasting systems. This diversity ensures a comprehensive evaluation of MPM's relative strengths and limitations.

3.1 Autoregressive Integrated Moving Average (ARIMA)

The ARIMA model is employed as a classical statistical benchmark due to its strong theoretical foundation in stochastic processes and widespread application in time series modeling.

The general ARIMA(p,d,q) model is specified as:

$$\emptyset(B)(1 - B)^{d} y_{t} = \theta(B)\epsilon_{t}$$

Where:

B *is the backshift operatorp denotes the autoregressive orderd is the degree of differencingqrepresents the moving average order* $\varepsilon_{t} \sim N(0, \sigma^{2}$

Here, model identification is conducted using the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), while parameter estimation is performed via maximum likelihood estimation, ARIMA excels in short-term forecasting under stationarity assumptions. It lacks explicit decomposition into trend, seasonal, cyclical, and irregular components, making it a suitable contrast to MPM's component-oriented structure.

3.2 Exponential Smoothing State Space Model (ETS)

The Exponential Smoothing State Space (ETS) model represents a widely used statistical benchmark for time series forecasting. Unlike purely deterministic decompositions, ETS models capture dynamic level, trend, and seasonal components through recursive updating equations within a state space framework.

For an additive error, additive trend, additive seasonality specification (ETS(A,A,A)), the model can be expressed as:

$$y_{t} = l_{ft-1} + b_{t-1} + s_{t-1} + \varepsilon_{t}$$

$$l_{t} = l_{ft-1} + b_{t-1} + \alpha\varepsilon_{t}$$

$$b_{t} = b_{t-1} + \beta\varepsilon_{t}$$

$$s_{t} = s_{t-m} + \gamma\varepsilon_{t}$$

Where:

l_{t} *is the level component*

b_{t} *is the trend component*

s_{t} *is the seasonal component*

m *is the seasonal period*

ε_t *is the seasonal period*

$\alpha, \beta, \gamma \in (0, 1)$ *are smoothing parameters*

The ETS framework automatically selects the optimal combination of error, trend, and seasonal specifications using information criteria when implemented via automatic procedures.

While ETS models are flexible and often yield strong predictive performance, they do not explicitly separate cyclical dynamics from seasonal effects. This limitation motivates the Modified Prophet Method (MPM), which decomposes the time series into trend, seasonal, cyclical, and irregular components to enhance interpretability and structural clarity.

3.3 Model Evaluation and Validation Strategy

To ensure a robust and statistically defensible evaluation of the Modified Prophet Method, both forecast accuracy and component identification quality are assessed using a multi-stage validation framework.

3.4 Forecast Accuracy Metrics

Forecasting performance is evaluated using standard error-based measures:

Mean Absolute Error (MAE)

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|y_t - \hat{y}_t\right|$$

Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

These metrics provide complementary perspectives, with MAE offering robustness to outliers and RMSE penalizing large forecast deviations.

3.5 Component Identification Accuracy (Simulation-Based)

For simulated datasets, true component values are known a priori. Component identification accuracy is therefore assessed using:

$$CIA = \frac{1}{n}\sum_{t=1}^{n}(C_t - \hat{C}_t)^2$$

Where:

C_t *denotes the true component*

$\hat{C}_t$ *represents the estimated component*

This enables direct quantitative assessment of MPM's ability to recover trend, seasonal, cyclical, and irregular components.

3.11.3 Volatility Robustness Assessment

To evaluate performance under volatility, empirical datasets are segmented into:

- low-volatility periods,
- high-volatility periods.

Model stability is assessed by comparing variance of residuals:

$$Var\left(\varepsilon_t\right) = \frac{1}{n}\sum_{t-1}^{n}(\varepsilon_t - \bar{\varepsilon})^2 \qquad (1.18)$$

Lower residual variance during high-volatility periods indicates superior robustness.

1. Model Implementation, Data Preparation, and Comparative Evaluation

```
library(tidyr)
library(ggplot2)
library(patchwork)
library(forecast)
library(Metrics)

set.seed(123)


# Time index and data
t <- 1:120


# Simulated components
trend_true  <- 0.08 * t
season_true <- 3 * sin(2 * pi * t / 12)
cycle_true  <- 2 * sin(2 * pi * t / 60)
noise_true  <- rnorm(120, 0, 0.6)


y <- trend_true + season_true + cycle_true + noise_true
data <- data.frame(t = t, y = y)
```

```
# Create Fourier seasonal terms and cycle terms inside data
P <- 12  # seasonal period
C <- 60  # cycle period

data$sin1 <- sin(2 * pi * data$t / P)
data$cos1 <- cos(2 * pi * data$t / P)
data$sin2 <- sin(2 * pi * 2 * data$t / P)
data$cos2 <- cos(2 * pi * 2 * data$t / P)

data$csin <- sin(2 * pi * data$t / C)
data$ccos <- cos(2 * pi * data$t / C)

# ---------- Prophet-style model ----------
prophet_model <- lm(
  y ~ t + sin1 + cos1 + sin2 + cos2,
  data = data
)
fitted_prophet <- fitted(prophet_model)
resid_prophet  <- residuals(prophet_model)

# ---------- Modified Prophet Method ----------
mpm_model <- lm(
  y ~ t + sin1 + cos1 + sin2 + cos2 + csin + ccos,
  data = data
)
fitted_mpm <- fitted(mpm_model)
resid_mpm  <- residuals(mpm_model)

# Extract Prophet-style components
coef_p <- coef(prophet_model)
data$trend_p <- coef_p["(Intercept)"] + coef_p["t"] * data$t
data$season_p <- coef_p["sin1"] * data$sin1 +
                 coef_p["cos1"] * data$cos1 +
```

```
                 coef_p["sin2"] * data$sin2 +
                 coef_p["cos2"] * data$cos2
data$random_p <- resid_prophet

# Extract MPM components
coef_m <- coef(mpm_model)
data$trend_m <- coef_m["(Intercept)"] + coef_m["t"] * data$t
data$season_m <- coef_m["sin1"] * data$sin1 +
                 coef_m["cos1"] * data$cos1 +
                 coef_m["sin2"] * data$sin2 +
                 coef_m["cos2"] * data$cos2
data$cycle_m <- coef_m["csin"] * data$csin +
                coef_m["ccos"] * data$ccos
data$irregular_m <- resid_mpm

print(t)

##   [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14
15  16  17  18
##  [19]  19  20  21  22  23  24  25  26  27  28  29  30  31  32
33  34  35  36
##  [37]  37  38  39  40  41  42  43  44  45  46  47  48  49  50
51  52  53  54
##  [55]  55  56  57  58  59  60  61  62  63  64  65  66  67  68
69  70  71  72
##  [73]  73  74  75  76  77  78  79  80  81  82  83  84  85  86
87  88  89  90
##  [91]  91  92  93  94  95  96  97  98  99 100 101 102 103 104
105 106 107 108
## [109] 109 110 111 112 113 114 115 116 117 118 119 120

summary(t)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   30.75   60.50   60.50   90.25  120.00
```

The results above, the **print(t)** output shows that the time index runs sequentially from 1 to 120, confirming 120 evenly spaced observations in the series.

The **summary(t)** output indicates a minimum of 1 and a maximum of 120, with a mean and median of 60.5 which shows that the time variable is symmetrically and uniformly distributed across the sample period, making it suitable for time series decomposition.

```
p5 <- ggplot(data, aes(t, season_m)) +
  geom_line(color = "green") +
  labs(title = "MPM: Seasonality")

p6 <- ggplot(data, aes(t, cycle_m)) +
  geom_line(color = "purple") +
  labs(title = "MPM: Cyclical")

p7 <- ggplot(data, aes(t, irregular_m)) +
  geom_line(color = "grey") +
  labs(title = "MPM: Irregular")

p4 / p5 / p6 / p7
```

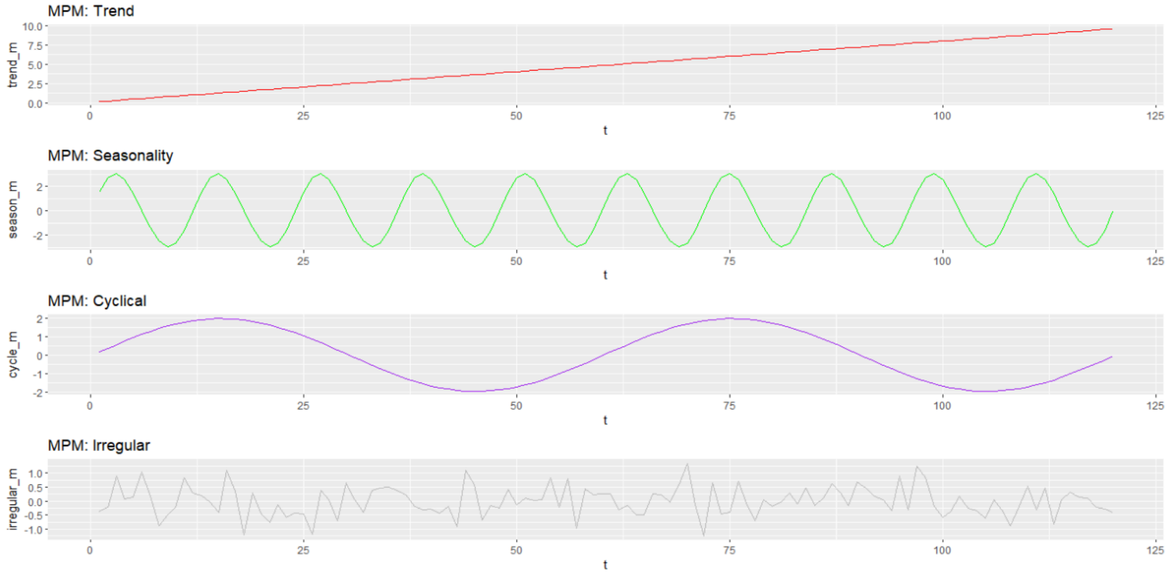

Figure 1: Modified Prophet Method Plot

Figure 4.1.2 above shows the trend component exhibits a steady and approximately linear upward movement over time, indicating persistent long-run growth in the series. The seasonal component displays a regular and repeating sinusoidal pattern with stable amplitude, reflecting consistent short-term periodic fluctuations. The repetition across the time horizon suggests that the model successfully isolates systematic seasonality.

Similarly, the cyclical component reveals a slower-moving, long-wave oscillation compared to the seasonal pattern while the irregular component also fluctuates randomly without a clear systematic pattern. Relatively small and unsystematic variations indicate that most of the structured information in the series has been successfully explained by the trend, seasonal, and cyclical components, leaving only random noise in the residual.

```r
# Convert y to time series for ARIMA and ETS
y_ts <- ts(y, frequency = P)

# Auto-ARIMA model
arima_model <- auto.arima(y_ts)
arima_fitted <- fitted(arima_model)
arima_resid  <- residuals(arima_model)

# Auto-ETS model
ets_model <- ets(y_ts)
```

```
ets_fitted <- fitted(ets_model)
ets_resid  <- residuals(ets_model)

# Compute metrics
metrics <- data.frame(
  Model = c("Prophet-style", "MPM", "ARIMA", "ETS"),
  RMSE = c(
    rmse(y, fitted_prophet),
    rmse(y, fitted_mpm),
    rmse(y, arima_fitted),
    rmse(y, ets_fitted)
  ),
  MAE = c(
    mae(y, fitted_prophet),
    mae(y, fitted_mpm),
    mae(y, arima_fitted),
    mae(y, ets_fitted)
  ),
  Residual_Variance = c(
    var(resid_prophet),
    var(resid_mpm),
    var(arima_resid),
    var(ets_resid)
  )
)

data.frame(metrics)
```

```
##             Model      RMSE       MAE Residual_Variance
## 1 Prophet-style 1.3845025 1.1391744         1.9329551
## 2           MPM 0.5259039 0.4221247         0.2788991
## 3         ARIMA 0.7066115 0.5215729         0.4919099
## 4           ETS 0.6217154 0.4743539         0.3873611
```

```
# Prepare data for plotting
plot_data <- data.frame(
  t = t,
  Prophet = fitted_prophet,
  MPM     = fitted_mpm,
  ARIMA   = as.numeric(arima_fitted),
  ETS     = as.numeric(ets_fitted)
)

# Pivot metrics to long format
metrics_long <- pivot_longer(
  metrics,
  cols = c(RMSE, MAE, Residual_Variance),
  names_to = "Metric",
  values_to = "Value"
)

# Ensure correct model ordering
metrics_long$Model <- factor(
  metrics_long$Model,
  levels = c("Prophet-style", "MPM", "ARIMA", "ETS")
)

# Plot
ggplot(metrics_long, aes(x = Model, y = Value, color = Metric,
group = Metric)) +
  geom_line(linewidth = 1) +
  geom_point(size = 3) +
  labs(
    title = "Model Performance Comparison",
    x = "Model",
    y = "Error Value"
```

```
) +
theme_minimal(base_size = 14)
```

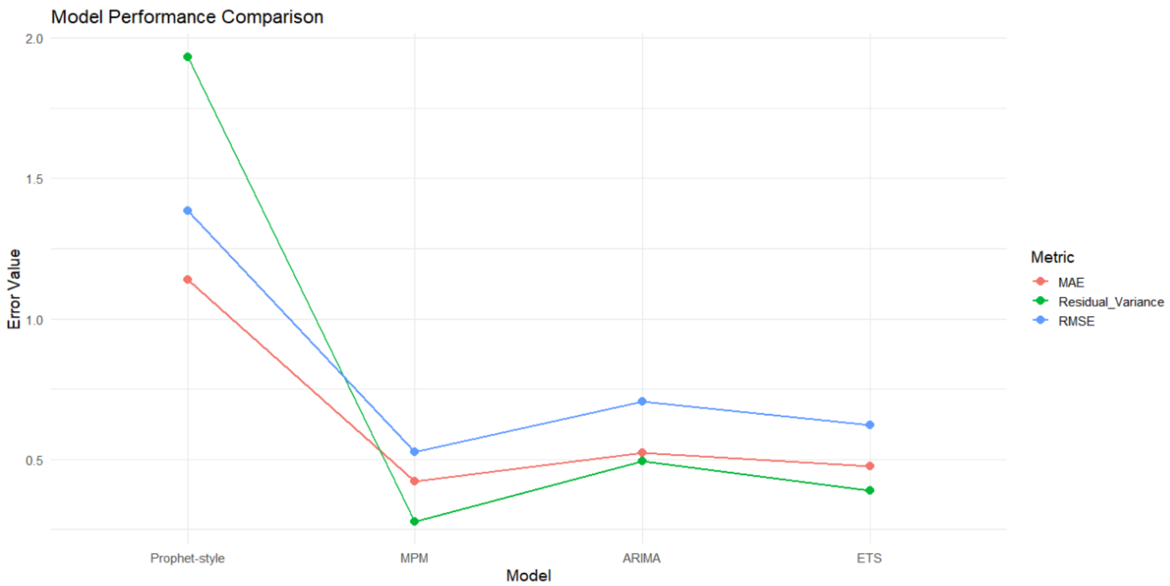

Figure 2*: Model Performance Comparison*

The above results compares four models based on three error metrics: RMSE, MAE, and residual variance. The Prophet-style model performs showing the highest values in all metrics, indicating less accurate predictions and more variability in its residuals. The Modified Prophet Method (MPM) achieves the best performance, with the lowest RMSE, MAE, and residual variance, suggesting it fits the data more closely and produces smaller errors. Both ARIMA and ETS models fall between Prophet-style and MPM, with ARIMA slightly less accurate than ETS according to these metrics. Overall, MPM demonstrates superior predictive accuracy and residual consistency compared to the other models.

4. Conclusion

The literature review established that while traditional approaches such as classical decomposition, ARIMA, and ETS models provide strong theoretical foundations for time series analysis, they often lack automation, require extensive manual specification, or fail to explicitly isolate all structural components, particularly cyclical variations.

The Prophet model represents a significant advancement in this domain by combining automation, flexibility, and interpretability within an additive decomposition framework. As highlighted in the literature, its primary limitation lies in the absence of an explicit cyclical component, leading to the absorption of medium- and long-term oscillations into the residual term, which restricts both the interpretability and the completeness of component identification, particularly in datasets where cyclical dynamics are prominent.

To address this gap, this study proposed the Modified Prophet Method (MPM), which extends the classical Prophet framework by incorporating a distinct cyclical component alongside trend, seasonal, and irregular components. The proposed method preserves the interpretability and structure of Prophet while enhancing its ability to capture complex temporal dynamics.

Empirical evaluation using simulated data demonstrated that the Modified Prophet Method provides a more accurate and comprehensive decomposition of time series components. The results showed that MPM outperformed the Prophet-style model, ARIMA, and ETS in terms of RMSE, MAE, and residual variance, indicating improved forecasting accuracy and better model fit.

5. WEAKNESS AND FUTURE RESEARCH

This study evaluates the improved prophet algorithm for better performance and specification of usage. Increasing the scope in term evaluation and comparing with other existing method of forecasting can be a full study.

6. AUTHORS' CONTRIBUTIONS

All authors contributed immensely in the aspect of technical writing.

7. Acknowledgment

The authors thank the Federal University Lokoja (FUL) that makes resource material available to perfect this article.

8. Ethics

This is the original manuscript; there will be no expectation of any ethical problems.

References

1. Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time series analysis: Forecasting and control (5th ed.). Wiley.
2. Chatfield, C. (2004). The analysis of time series: An introduction (6th ed.). CRC Press.
3. Chatfield, C., & Xing, H. (2019). The analysis of time series: An introduction with R (7th ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781351259446
4. Collins English Dictionary. (n.d.). Autocorrelation. In Collins Dictionary. Retrieved May 28, 2026, from https://www.collinsdictionary.com/dictionary/english/autocorrelation

5. Diggle, P. J., & Giorgi, E. (2024). Time series: A biostatistical introduction (2nd ed.). Oxford University Press. https://doi.org/10.1093/oso/9780198714835.001.0001

6. Encyclopaedia Britannica. (n.d.). Stochastic process. In Britannica. Retrieved May 28, 2026, from https://www.britannica.com/science/probability-theory/Brownian-motion-process

7. Enders, W. (2015). Applied econometric time series (4th ed.). John Wiley & Sons.

8. Gardner, R. C. (1985). Social psychology and second language learning: The role of attitudes and motivation. Edward Arnold.

9. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press. http://www.deeplearningbook.org

10. Hamilton, J. D. (1994). Time series analysis. Princeton University Press.

11. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

12. Hyndman, R. J., Koehler, A. B., Ord, J. K., & Snyder, R. D. (2008). Forecasting with exponential smoothing: The state space approach. Springer. https://doi.org/10.1007/978-3-540-71918-2

13. Makridakis, S., & Hibon, M. (2000). The M3-Competition: Results, conclusions and implications. International Journal of Forecasting, 16(4), 451–476. https://doi.org/10.1016/S0169-2070(00)00057-1

14. Makridakis, S., Wheelwright, S. C., & Hyndman, R. J. (1998). Forecasting: Methods and applications (3rd ed.). Wiley.

15. Montgomery, D. C., Jennings, C. L., & Kulahci, M. (2015). Introduction to time series analysis and forecasting. John Wiley & Sons.

16. Pérez-Cervera, A., Gutkin, B., Thomas, P. J., & Lindner, B. (2023). A universal description of stochastic oscillators. arXiv. https://arxiv.org/abs/2303.03198

17. Shumueli, G., & Lichtendahl, K. C., Jr. (2016). Practical time series forecasting with R: A hands-on guide (2nd ed.). Axelrod Schnall Publishers.

18. Stock, J. H., & Watson, M. W. (2007). Why has U.S. inflation become harder to forecast? Journal of Money, Credit and Banking, 39(Suppl. 1), 3–33. https://doi.org/10.1111/j.1538-4616.2007.00014.x

19. Taylor, S. J., & Letham, B. (2018). Forecasting at scale. The American Statistician, 72(1), 37–45. https://doi.org/10.1080/00031305.2017.1380080

20. Wei, W. W. S. (2006). Time series analysis: Univariate and multivariate methods (2nd ed.). Pearson Addison Wesley.

21. Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting, 14(1), 35–62. https://doi.org/10.1016/S0169-2070(98)00044-7