Original Research Article

Enhancing Agricultural Commodity Forecasting: A Median-Based Combination of Time Series Models

Abstract

This study explores the efficacy of combining forecasts using the median operator to enhance forecasting performance. The traditional approach of assigning equal weights to individual models often struggles with extreme forecasts. A new methodSimple Combination of Univariate Models (SCUM) is utilized, which uses the median operator to combine forecasts from four distinct time series models: Exponential Smoothing (ETS), Auto Regressive Integrated Moving Average (ARIMA), Dynamically Optimised Theta Model (DOTM), and Complex Exponential Smoothing (CES). This approach aims to mitigate the influence of extreme forecasts and improve overall accuracy.

Our empirical analysis investigates the use of the SCUM approach for the agricultural commodity data. Yearly production of Rice is used, sourced from the Ministry of Agriculture & Farmers Welfare, Government of India. The forecasting performance of the SCUM approach is compared against individual models and a mean-based combined forecast using key performance metrics such as Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and Relative Efficiency. The results show that SCUM outperforms all other models, achieving the lowest RMSE (51.89) and a MAPE of 6.90, indicating superior in-sample forecasting accuracy. ARIMA was the least efficient with a 22.11% relative efficiency, while the Mean combination method (3.33%) was closest to SCUM in performance. The findings suggest that SCUM is not only a viable alternative to traditional methods but also offers significant advantages in improving forecasting accuracy. This research underscores the potential for median-based forecast combinations to achieve superior predictive accuracy and reliability, making it a valuable tool for scholars and practitioners in time series forecasting.

Keywords: ETS; ARIMA; DOTM; CES; Median; Relative Efficiency

1. Introduction

Quantitative data collected over successive time intervals constitute time series data. Examples include monthly, quarterly, or yearly records of consumption and production of various commodities. Time series analysis helps in understanding the underlying forces that drive trends and in forecasting by fitting appropriate models. This type of analysis is instrumental in numerous fields, such as planning new economic schemes and improving agriculture, health, and other sectors. Although various models have been developed for forecasting, the primary knowledge gap identified is the absence of a simple yet effective approach to enhance forecasting accuracy. Traditional methods often rely on individual models, which can result in suboptimal performance due to their limitations in capturing the full complexity of time series data. While more sophisticated techniques exist, they are often computationally intensive and difficult to implement in practical applications. This highlights a clear need for a solution that balances accuracy with computational efficiency. Addressing this gap, the literature underscores the necessity for a method that can effectively integrate the strengths of various forecasting models while remaining accessible and practical for widespread use. This issue can be mitigated by combining multiple candidate models, a method known as combined forecasting.

The initial efforts in combined forecasting employed general analytical models (Reid, 1968; Bates and Granger, 1969), which expanded the theoretical foundation for this approach. Early combined forecasts included Bayesian combinations (Dickinson, 1973), minimum variance models (Bunn, 1975), and approaches that consider the appropriateness of different probabilities for combining forecasts (French, 1980; Bunn, 1981). These combined forecasts can outperform the best single component forecast (Bates and Granger, 1969). Combination weights can be equally distributed or proportionally adjusted based on past model errors, with numerous sophisticated combination schemes proposed in the literature. For instance, Granger and Ramanathan (1984) suggested the use of unconstrained and negative weights.

Hibon and Evgeniou (2005) explored the trade-offs between combining forecasts and the risk of incorrect model selection, concluding that combining forecasts can help avoid significant errors from incorrect model selection. The arithmetic mean combination is susceptible to extreme values; thus, trimmed means are used to reduce extreme errors (Jose and Winkler, 2008). Attention has also been directed towards alternative strategies, such as using the median and mode, as well as trimmed and winsorized means (Genre et al., 2013; Jose et al., 2014; Grushka-Cockayne et al., 2017), due to their robustness and lower sensitivity to extreme forecasts compared to simple averages (Lichtendahl and Winkler, 2020).

Kourentzes et al. (2014) conducted empirical comparisons of mean, mode, and median combination operators using kernel density estimation. Their findings indicated that these operators handle extreme values differently, with the mean being the most sensitive and the mode the least. These results have led to calls for further research into the use of mode and median operators, which have been underexplored in the literature. Even in the big data era, where sophisticated combination methods and machine learning algorithms are prevalent, simple combinations remain relevant. An example is the M-competition, the first forecasting competition organized by Spyros Makridakis and Michele Hibon, which involved 1001 time series (Hyndman, 2020). The recent M4 competition (Makridakis et al., 2020) showed that simple combinations still achieve competitive forecasting performance. Specifically, a simple equal-weight combination ranked third for yearly time series (Shaub, 2019), and a median combination of four basic forecasting models placed sixth for point forecasting (Petropoulos and Svetunkov, 2020).

2. Methodology

2.1. Exponential Smoothing (ETS) Model

Exponential smoothing involves forecasting by assigning exponentially decreasing weights to past observations, making recent data more influential in the forecast. Various methods of exponential smoothing have evolved since its inception. These include Simple Exponential Smoothing (Brown, 1957), Holt's Exponential Smoothing for trend analysis (Holt, 1975), and Holt-Winter's Exponential Smoothing for capturing both trend and seasonality (Holt, 1957; Winters, 1960)

2.1.1. Simple Exponential Smoothing (Brown, 1959)

Simple Exponential Smoothing is applied to time series data that lacks any trend or seasonal pattern. It can be represented as follows:

Forecast equation: $\hat{y}_{t+\;|t} = l_t$

Smoothing equation: $l_t \;=\; \alpha\, y_t + (1 \;-\; \alpha) l_{t-1}$

where α is the smoothing parameter, $0 \leq \alpha \leq 1$

2.1.2. Holt’s Exponential Smoothing for trend (Holt, 1957)

Holt’s Exponential Smoothing is utilized for time series data exhibiting a linear trend without seasonal effects. This method is characterized by a forecast equation along with two smoothing equations (one for level and one for trend):

Forecast equation: $\hat{y}_{t+\ \ |t} = l_t + \quad b_t$

Level equation: $l_t = \alpha\, y_t + (1 - \alpha)(l_{t-1} + b_{t-1})$

Trend equation: $b_t = \beta(\, l_t - l_{t-1}) + (1 - \beta)b_{t-1}$

where α is the smoothing parameter, $0 \leq \alpha \leq 1$; β is the trend parameter, $0 \leq \beta \leq 1$.

2.1.3. Holt-Winter’s Exponential Smoothing for trend and seasonality (Holt, 1957; Winters 1960)

Holt (1957) and Winters (1960) enhanced Holt’s exponential smoothing method to account for seasonal patterns in time series data. The Holt-Winters Exponential Smoothing method is defined by the following forecast equation along with three additional equations (level, trend, and seasonal equations):

Forecast equation: $\hat{y}_{t+\ \ |t} = l_t + \quad b_t + s_{(t-m+\ \ ^{+}_{m})}$

Level equation: $l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t+1} + b_{t+1})$

Trend equation: $b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1}$

Seasonal equation: $s_t = \gamma(y_t - l_t - b_{t-1}) + (1 - \gamma)s_{t-s}$

where α is the smoothing parameter, $0 \leq \alpha \leq 1$; where β is the smoothing parameter, $0 \leq \beta \leq 1$; where γ is the smoothing parameter for seasonality, $0 \leq \gamma \leq 1$. m denotes the order or frequencyof the seasonality and $\ \ ^{+}{}_{m} = [(\mathrm{h} - 1)\mathrm{mod\ m}] + 1$.

Exponential smoothing methods were initially categorized by Pegels (1969). Subsequent modifications and extensions were made by Gardner (1985), Hyndman et al. (2002), and Taylor (2003). This classification is primarily based on the trend and seasonality patterns observed in the time series. The trend component can fall into one of five categories: absent (N), additive (A), additive damped (Ad), multiplicative (M), or multiplicative damped (Md). Similarly, the seasonal component can be categorized as absent (N), additive (A), or multiplicative (M). Altogether, these classifications result in fifteen distinct exponential smoothing methods documented in the literature.

Table 1: Types of exponential smoothing methods

Trend component	**Seasonal component**		
	N	**A**	**M**
N	N,N	N,A	N,M
A	A, N	A, A	A, M
Ad	Ad, N	Ad, A	Ad, M
M	M, N	M, A	M, M
Md	Md, N	Md, A	Md, M

Hyndman et al. (2008) presented two potential innovations for state space models, based on the error term being either additive or multiplicative, for each of the fifteen exponential smoothing models. This resulted in a total of thirty different ETS models. Additionally, Hyndman (2002) developed an automatic forecasting method to accommodate these thirty ETS models.

Table 2: Additive ETS Model

Trend component	**Seasonal component**		
	N	**A**	**M**
N	N,A, N	N,A, A	N,A, M
A	A,A, N	A,A, A	A,A, M
Ad	A,Ad, N	A,Ad, A	A,Ad, M
M	A,M, N	A,M, A	A,M, M
Md	A,Md, N	A,Md, A	A,Md, M

Table 3: Multiplicative ETS Model

Trend component	**Seasonal component**		
	N	**A**	**M**
N	M,A, N	M,A, A	M,A, M
A	M,A, N	M,A, A	M,A, M
Ad	M,Ad, N	M,Ad, A	M,Ad, M
M	M,M, N	M,M, A	M,M, M
Md	M,Md, N	M,Md, A	M,Md, M

2.2. ARIMA model (Box and Jenkins, 1976)

The Auto-Regressive Moving Average (ARMA) model can be extended to the ARIMA model by incorporating differencing. The ARMA model itself is a combination of an autoregressive model and a moving average model. Thus, the ARIMA model includes past values and past error terms, along with the differencing level of the dataset. Differencing is essential to ensure the series is stationary, which is a crucial assumption in time series modeling. The ARIMA model is denoted as ARIMA(p, d, q), where 'p' represents the order of the autoregressive model, 'd' indicates the degree of differencing, and 'q' specifies the order of the moving average model. Therefore, an ARIMA model can be expressed as follows:

$$\phi(B)(1 - \mathrm{B})^d Y_t = \theta(B)\varepsilon_t$$

where, $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p$ is the autoregressive operator of order p, with $\phi_1, \phi_2, \ldots, \phi_p$ as the corresponding autoregressive parameters; $\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q$ is the moving average operator of order q, with $\theta_1, \theta_2, \ldots, \theta_q$ as the associated moving average parameters; and $(1 - \mathrm{B})^d$ is the differencing operator of order d produce stationarity of the d^t differenced data.In this ae model equation, B is used as a backshift operator on Y_t and is defined as $B^i(Y_t) = Y_{t-1}$.

Analyzing time series data with an ARIMA (p, d, q) model involves the following steps:

i. **Observe the Dataset and Ensure Stationarity**: Begin by examining the internal components of the time series data, such as trend, seasonality, and cycles. The dataset should be checked for stationarity. If the data is non-stationary, use the differencing method to make it stationary. Typically, first differencing is sufficient, but in some cases, second differencing may be necessary to achieve stationarity.

ii. **Identification**: Identify the order of the model parameters p, d, and q. The differencing level d is straightforward, as it corresponds to the number of differences applied to achieve stationarity. Use the autocorrelation function (ACF) and partial autocorrelation function (PACF) to determine the order of the autoregressive (AR) and moving average (MA) components. If the ACF exponentially declines to zero, the significant lags in the PACF indicate the AR parameters. Conversely, if the PACF exponentially declines to zero, the significant lags in the ACF indicate the MA parameters.

iii. **Estimation:** Precisely estimate the model parameters using the method of least squares.

iv. **Diagnostic Checking:** Select the best-fitting ARIMA model using various model selection criteria, such as Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC). The model with the lowest value of these statistics is considered the best fit. Additionally, perform diagnostic checks on the residuals of the model using the Ljung-Box test:

$$Q^* = n(n+2)\sum_{k=1}^{h}(n-k)^{-1}r_k^2$$

where h is the maximum lag being considered and n is the number of observations in the series.

v. **Forecasting:** Conduct short-term forecasting, typically up to 12 points ahead.

2.3. Complex Exponential Smoothing (Svetunkov and Kourentzes, 2015)

In Complex Exponential Smoothing, two key characteristics are proposed instead of decomposing the time series: i) the observed value of the series; and ii) information potential, a non-observable component that influences the observed values and contains additional useful information about the series. This information potential enhances model flexibility, enabling the capture of a wider range of behaviors, eliminating the need for an arbitrary distinction between level and trend, and resulting in increased forecasting accuracy. Complex Exponential Smoothing can be represented as follows:

$$\hat{y}_{t+1} + i\hat{p}_{t+1} = (\alpha_0 + i\alpha_1)(y_t + ip_t) + (1 - \alpha_0 + 1 - i\alpha_1)(\hat{y}_t + i\hat{p}_t)$$

where, $\hat{y}_t$is the forecast of the actual series

$\hat{p}_t$ is the estimate of the information potential

$\alpha_0 + i\alpha_1$ is complex smoothing parameter

Representing the complex-valued function as a system of two real-valued functions as:

$$\hat{y}_{t+1} = (\alpha_0 y_t + (1-\alpha_0)\hat{y}_t) - (\alpha_1 p_t + (1-\alpha_1)\hat{p}_t)$$
$$\hat{p}_{t+1} = (\alpha_1 y_t + (1-\alpha_1)\hat{y}_t) - (\alpha_0 p_t + (1-\alpha_0)\hat{p}_t)$$

The state-space model of CES used to further explore its properties given by:

$$y_t = l_{t-1} + \varepsilon_t$$
$$l_t = l_{t-1} - (1-\alpha_1)c_{t-1} + (\alpha_0 - \alpha_1)\varepsilon_t$$
$$c_t = l_{t-1} + (1-\alpha_0)c_{t-1} + (\alpha_0 + \alpha_1)\varepsilon_t$$

where, l_t is the level component

c_t is the information component on observation t

$\varepsilon_t \sim N(0, \sigma^2)$.

2.4.1. Theta method (Assimakopoulos*et al.*, 2000)

The Theta model is based on the concept of adjusting the local curvatures of the time series as follows:

$$\nabla^2 Z_t(\theta) = \theta \nabla^2 Y_t$$

where $Y_1, \dots, Y_n$ is the original time series (non-seasonal or deseasonalised), and ∇ is the difference operator. The initial values of Z_1 and Z_2 are obtained by minimising $\sum_{t=1}^{n}[\mathrm{Y}_t - \mathrm{Z}_t(\theta)]^2$. An analytical solution to compute Z(θ) (Hyndman et al., 2003) is given by

$$\mathrm{Z}_t(\theta) = \theta \mathrm{Y}_t + (1-\theta)(\mathrm{A}_n + \mathrm{B}_n t), t = 1, \dots, n$$

where A_n and B_n are the minimum square coefficients of a simple linear regression over $Y_1, \dots, Y_n$ against 1, … , n, given by

$$\mathrm{A}_n = \frac{1}{n}\sum_{t=1}^{n} \mathrm{Y}_t - \frac{\mathrm{n}+1}{2}\mathrm{B}_n;$$

$$\mathrm{B}_n = \frac{6}{\mathrm{n}^2 - 1}\left(\frac{2}{n}\sum_{t=1}^{n} t\mathrm{Y}_t - \frac{1+\mathrm{n}}{n}\sum_{t=1}^{n}\mathrm{Y}_t\right)$$

2.4.2. DOTM (Fiorucci *et al.*, 2016)

For A_t and B_t as fixed coefficients for all t.these coefficients as dynamic functions. This means that for updating the state t to t +1 ,only consider the prior information $Y_1, \dots, Y_t$used when computing A_t and B_t. Therefore, state space framework for DOTM is given by:

$$y_t = \mu_t + \varepsilon_t$$

$$\mu_{t-1} = l_{t-1} + \left(1 - \frac{1}{\theta}\right)\left\{(1-\alpha)^t \mathrm{A}_t + \left[\frac{1-(1-\alpha)^{t+1}}{\alpha}\right]\mathrm{B}_t\right\}$$

$$l_t = \alpha Y_t + (1-\alpha) l_{t-1}$$

$$\mathrm{A}_t = \overline{\mathrm{Y}}_t - \frac{\mathrm{t}+1}{2}\mathrm{B}_t$$

$$\mathrm{B}_t = \frac{1}{t+1}[(\mathrm{t}-2)\mathrm{B}_{t-1} + \frac{6}{t}(\mathrm{Y}_t - \overline{\mathrm{Y}}_{t-1})$$

$$\overline{\mathrm{Y}}_t = \frac{1}{t}[(t-1)\overline{\mathrm{Y}}_{t-1} + \mathrm{Y}_t]$$

where, μ_t is the mean component

l_t is the level component
A_t and $B_t$are dynamic square coefficient
$\varepsilon_t \sim N(0, \sigma^2)$.

The forecast equation at $h = 1$ is given by

$$\hat{Y}_{t+1|t} = l_t + \left(1-\frac{1}{\theta}\right)\left\{(1-\alpha)^t A_t + \left[\frac{1-(1-\alpha)^{t+1}}{\alpha}\right] B_t\right\}$$

2.5. Simple Combination of Univariate Models (SCUM)(Petropoulos and Svetunkov, 2020)

This approach utilizes the median to combine forecasts, as illustrated in Figure 1. Models that produce the most extreme forecasts are discarded, and the median of the point forecasts from the four models is taken as the final combined forecast. If the pool of models is small, the difference between the final forecast and those obtained from individual models is minimal. This method reduces the influence of models that perform poorly for certain time series. For example, if the ARIMA model produces a forecast with a downward trend while other models predict an upward trend, the final forecast derived from the median will reflect an upward trend.

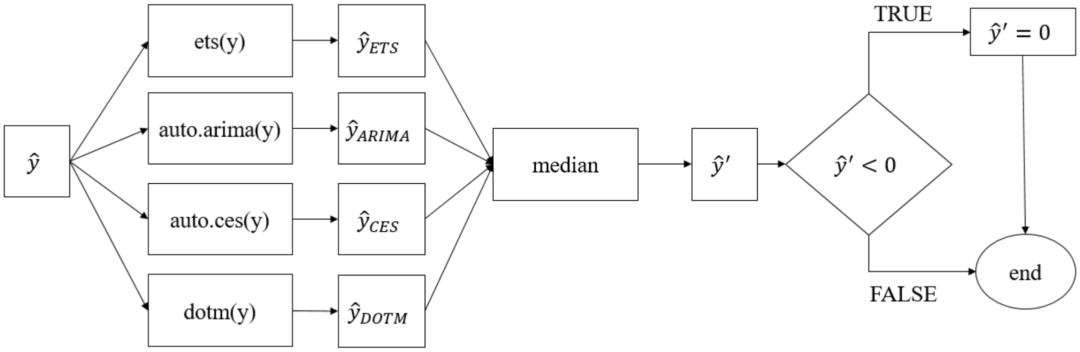

Figure 1: Flowchart of SCUM model

3. Results and Discussion

3.1. Data Description

To illustrate the discussed models, the annual production data of rice, an agricultural commodity produced in India, was analyzed. The data was sourced from the Ministry of Agriculture & Farmers Welfare, Government of India. The dataset comprises 58 data points, covering the years 1961-62 to 2018-2019.

Table 4: Summary statistics of the datasets

Statistics	**Value**
Minimum(in Lakh Tonnes)	304.40
Maximum(in Lakh Tonnes)	1164.20
Mean(in Lakh Tonnes)	693.80
Median(in Lakh Tonnes)	723.40
Standard Deviation(in Lakh Tonnes)	253.94
Coefficient Of Variation	0.36
Kurtosis	-1.34
Skewness	0.10

Table 4 provides the summary statistics of the dataset. The summary statistics reveal that the annual production of rice in India has a minimum value of 304.40 lakh tonnes and a maximum value of 1164.20 lakh tonnes, with a mean of 693.80 lakh tonnes. The median value of 723.40 lakh tonnes indicates a slight skew towards higher production values, as supported by the positive skewness of 0.10. The dataset exhibits a moderate level of variation with a standard deviation of 253.94 lakh tonnes and a coefficient of variation of 0.36. The negative kurtosis value of -1.34 suggests a relatively flat distribution compared to a normal distribution.

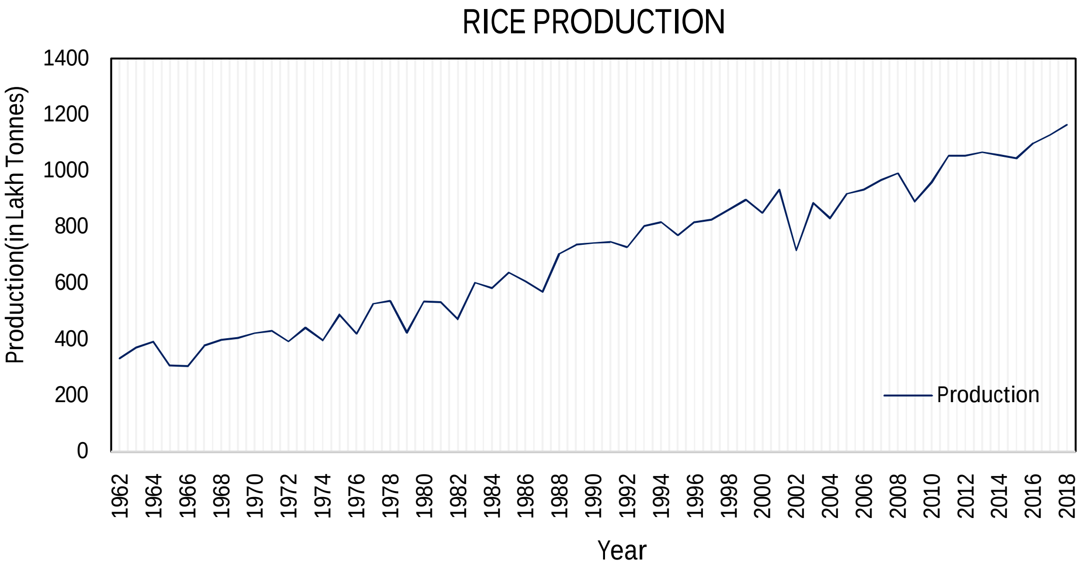

Figure 2 Line Graph of the dataset

Figure 2 displays line graph of the annual rice production in India from 1961-62 to 2018-2019. The line graph shows an overall increasing trend in rice production over the years, with some noticeable fluctuations. This visual representation provides an essential context for understanding the underlying patterns and trends in the data, which will be crucial for modeling and forecasting.

3.2. Test for Stationarity

The stationarity of data is a crucial property of time series analysis. To determine whether the dataset is stationary, three statistical tests were employed: the Augmented Dickey–Fuller (ADF) test, the Phillips-Perron (PP) test, and the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test.The null hypothesis of the ADF test and PP test posits that the dataset is non-stationary. The null hypothesis of the KPSS test posits that the dataset is stationary.

Table 5: Test for stationarity

	Dataset		
	ADF test	**PP test**	**KPSS Test**
Test Statistics	-3.77	-50.96	0.12
p-value	0.02	0.01	0.10

From Table 5, the ADF test, PP test, and KPSS test suggest that the dataset is stationary for the purposes of further analysis.

3.3. Test for Nonlinearity: The nonparametric Broock-Dechert-Scheinkman (BDS) test (Broock et al., 1996) is utilized to assess the presence of nonlinearity in the dataset. This test examines whether the data exhibit nonlinear dependence by testing the null hypothesis that the residuals are independently and identically distributed. A rejection of the null hypothesis would suggest nonlinearity. Table 6 presents both the BDS test statistic values and the corresponding p-values (in parentheses). As shown, all p-values are greater than 0.01, indicating that the null hypothesis is not rejected. This suggests that the dataset does not exhibit significant nonlinearity and may follow a linear pattern.

Table 6: BDS test results

Dimension (m)	0.5σ	1.0σ	1.5σ	2.0σ
2	1.1775 (0.019)	1.3684 (0.013)	-0.3402 (0.641)	0.6123 (0.901)
3	1.0732 (0.121)	0.8700 (0.184)	1.2788 (0.011)	0.5338 (0.821)

3.4. ARIMA model

Following the detection of stationarity and linearity in the dataset through the earlier tests, the dataset was deemed suitable for fitting an ARIMA model. The selection of the ARIMA model was guided by analyzing the ACF and PACF plots, which were applied to the dataset. The ACF plot, showing the correlation between the time series and its lagged values, exhibited an exponential decay pattern, as depicted in Figure 3. Meanwhile, the PACF plot, which highlights the correlation between the time series and its lagged values after removing the influence of intermediate lags, showed a significant spike at lag 1, followed by near-zero correlations at higher lags, also illustrated in Figure 3. These characteristic patterns in the ACF and PACF plots confirmed that an ARIMA (1,0,0) model is appropriate for this dataset.

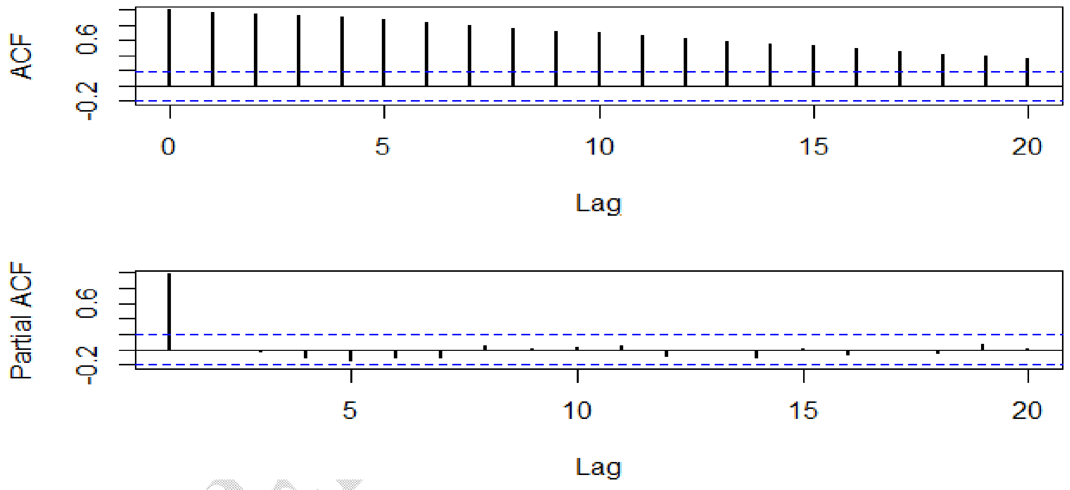

Figure 3ACF and PACFplots of the dataset

Based on these observations, the ARIMA (1, 0, 0) model was selected as the best-fitting model. The model's selection was further validated using the AIC and BIC values, with the ARIMA(1, 0, 0) model presenting the lowest AIC value of 661.40 and a BIC value of 664.25. The parameter estimates for the ARIMA (1, 0, 0) model are detailed in Table 7. Additionally, the actual and fitted values for this model are displayed in Figure 4. The ARIMA model was implemented using the Arima() function available in the forecast 8.11 package.

Table 7: Parameters estimate of ARIMA(1,0,0) model

Parameters	**AR1**

Estimate	0.98
Standard Error	0.02
AIC value	661.40
BIC value	664.25

The Ljung-Box test was applied to evaluate the presence of serial autocorrelation in the residuals from the ARIMA(1, 0, 0) model. The null hypothesis of the test suggests no autocorrelation in the residuals for a specified number of lags. The test results indicated no significant autocorrelation, with a test statistic of -3.77 and a p-value of 0.12.

3.5. Exponential Smoothing (ETS) Model

The implementation of the Exponential Smoothing (ETS) model was done using the `smooth` 2.5.5 package with the `ets()` function. The model selected from this package is ETS (A,A,N), known as Holt's Exponential Smoothing for trend. The parameter estimates of the ETS model are provided in Table 8, while the actual and fitted values of the ETS model are shown in Figure 4.

Table 8: Parameters of the ETS model

Model: ETS (A, A, N)	
$\boldsymbol{\alpha}$	0.24
$\boldsymbol{\beta}$	0.0001
AIC value	700.87

3.6. CES Model

Implementation of Complex Exponential Smoothing (CES) Model is done by *smooth 2.5.5* package by auto.ces() function. The parameters estimate of CES model are provided in the Table 9,while the actual and fitted values of the CES model are shown in Figure 4.

Table 9: Parameters of the CES model

Parameter	Value
$\boldsymbol{\alpha_0}$	1.32
$\boldsymbol{\alpha_1}$	1.02
AIC value	628.05

3.7. DOTM Model

Implementation of Dynamically Optimized Theta Model (DOTM) is done by *forecTheta 2.2* package by dotm() function. The parameters estimate and of DOTM model are provided in the following Table 10,while the actual and fitted values of the DOTM model are shown in Figure 4.

Table 10: Parameters of the DOTM model

Parameter	Value
$\boldsymbol{l_0}$	51.28
$\boldsymbol{\alpha}$	0.27
$\boldsymbol{\theta}$	1232255.86
AIC value	632.60

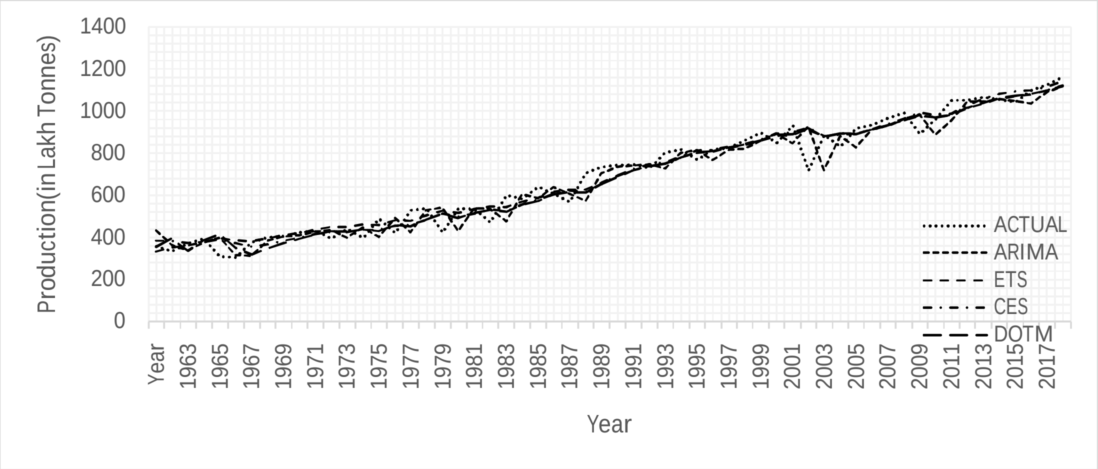

Figure 4: Fitted Graph of Candidate Models

3.8. Mean Combination

A mean combined forecast is obtained in which weights are given using arithmetic mean. The Weights in the combined forecasts are equally assigned to each model. The fitted value of all the candidate model based on the original time series data are estimated. The fitted value of their respective candidate model is combined by using arithmetic mean. The mean used for combination is implemented by apply() function.The actual and fitted value of mean combination is shown in the figure 5.

3.9. SCUM (Median Combination)

A combined forecast is obtained in which median is used to combined the forecasts. The fitted value of all the candidate model based on the original time series data are estimated. The fitted value of their respective candidate model is combined by using median. The median used for combination is implemented by apply() function. After implementation if the fitted value has value less than zero, then assigned the value to zero.The actual and fitted value of SCUM is shown in the figure 5.

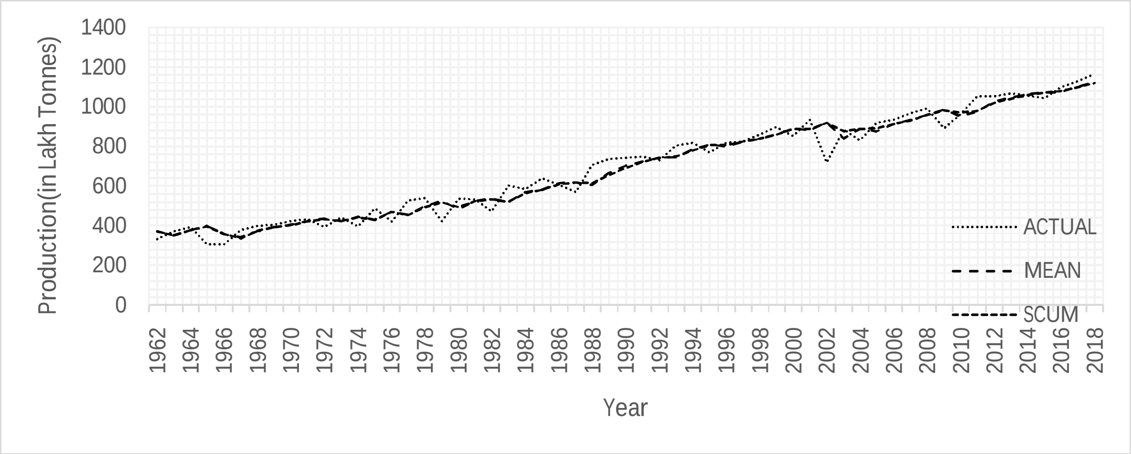

Figure 5: Fitted Graph of Mean combination and SCUM Models

The Table 11 lists the forecasting models used in the analysis along with the corresponding R packages and functions utilized for their implementation:

Table 11: Forecasting models and the corresponding R functions.

Model	**R package**	**Function**
ARIMA	*forecast 8.11*	Arima()
ETS	*forecast 8.11*	ets()
CES	*smooth 2.5.5*	ces()
DOTM	*forecTheta 2.2*	dotm()

3.10. Performance Measures

The forecasting performance of the SCUM forecasts have been compared with their counterparts using threeevaluation measures:

Root Mean Squared Error (RMSE): This metric measures the average magnitude of the errors between the predicted and actual values. Lower RMSE values indicate better model performance. RMSE is calculated by

$$RMSE = \sqrt{\frac{1}{h}\sum_{t=1}^{h}(e_t)^2}$$

where $e_t = y_t - \hat{y}_t$ and h is the forecast horizon

Mean Absolute Percentage Error (MAPE):MAPE measures the average absolute percentage error between the predicted and actual values. Similar to RMSE, lower MAPE values indicate better model performance. MAPE is calculated by

$$MAPE = \frac{1}{h}\sum_{t=1}^{h}|e_t|/y_t \times 100$$

where $e_t = y_t - \hat{y}_t$ and h is the forecast horizon

Relative Efficiency: This metric compares the efficiency of each model relative to SCUM, with SCUM set as the benchmark (0.00%). The lower the relative efficiency percentage, the closer the model's performance is to that of SCUM.A positive value of Relative Efficiency show that SCUM is better than other model. Relative Efficiency of SCUM is calculated by

$$\text{Relative Efficiency} = \frac{MAPE_{Model} - MAPE_{SCUM}}{MAPE_{SCUM}} \times 100$$

These performance measures provide a comprehensive assessment of the forecasting accuracy and efficiency of the SCUM method compared to other models.

The Table 12 compares the in-sample performance of various univariate forecasting models and combined forecast methods. The models compared include ARIMA, ETS, CES, and DOTM, as well as two combined forecast methods: the mean combination and the SCUM.

Table 12: Comparisons among the models (in sample)

	Univariate Models				**Combined Forecasts**	
	ARIMA	**ETS**	**CES**	**DOTM**	**Mean**	**SCUM**
RMSE	66.92	60.68	55.60	53.63	52.92	51.89
MAPE	8.42	7.44	7.22	7.40	7.13	6.90

Relative Efficiency (in %)(SCUM)	22.11	7.82	4.64	7.30	3.33	0.00

According to Table 12, SCUM has the lowest RMSE (51.89), indicating it provides the most accurate in-sample predictions among the models. SCUM also demonstrates the best performance with a MAPE of 6.90, followed closely by the Mean combination (7.13) and CES (7.22). ARIMA has the highest relative efficiency percentage (22.11%), indicating it is the least efficient compared to SCUM, while the Mean combination method (3.33%) is closest in performance to SCUM. Overall, the SCUM method demonstrates superior performance in terms of both RMSE and MAPE, making it the most efficient and accurate model for in-sample forecasting in this comparison.

4. Conclusion

This study uses the SCUM approach, which leverages the median operator to combine forecasts from multiple univariate time series models, specifically ETS, ARIMA, DOTM, and CES models. The SCUM method aims to provide a simple yet effective approach by combining the forecasts of multiple univariate time series models, thereby enhancing overall forecasting accuracy and mitigating the impact of extreme forecasts. Through rigorous empirical analysis using annual rice production data from India, the performance of SCUM is benchmarked against individual models and mean-based combined forecasts. The results demonstrate that SCUM outperforms all other models, achieving the lowest RMSE (51.89) and a MAPE of 6.90, indicating its robustness and effectiveness. By using the median operator, SCUM successfully reduces the influence of extreme forecasts, a common drawback of mean-based combinations. The results show that ARIMA was the least efficient, with a 22.11% relative efficiency, while the Mean combination method (3.33%) was closest to SCUM in performance, with the SCUM model serving as the benchmark model. The use of common R packages makes SCUM easily adoptable in existing workflows. The contributions and findings of this study pave the way for future advancements in model combination strategies, offering valuable insights for both academic researchers and industry practitioners.

Future research could test SCUM across diverse datasets to validate its generalizability. Additionally, exploring advanced techniques to optimize the combination weights dynamically, potentially incorporating machine learning and deep learning algorithms, could further enhance the method. Extending the analysis to long-term forecasting horizons is also suggested to assess the stability and reliability of the SCUM method over extended periods.

Disclaimer (Artificial intelligence)
Option 1:
Author(s) hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc) and text-to-image generators have been used during writing or editing of manuscripts.
Option 2:

Author(s) hereby declare that generative AI technologies such as Large Language Models, etc have been used during writing or editing of manuscripts. This explanation will include the name, version, model, and source of the generative AI technology and as well as all input prompts provided to the generative AI technology

Details of the AI usage are given below:
1.Author(s) hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc) and text-to-image generators have been used during writing or editing of manuscripts.

2.
3.

References

Assimakopoulos, V. and Nikolopoulos, K. (2000). The theta model: a decomposition approach to forecasting. *International Journal of Forecasting*, **16**(4), 521–530.

Bates, J. M.and Granger, C. W. J. (1969). The combination of forecasts. *The Journal of the OperationalResearch Society*,**20**(4), 451–468.

Box, G.E. and Jenkins, G.M. (1976). *Time series analysis: Forecasting and control*, San Francisco.

Brown, R. G. (1959). *Statistical forecasting for inventory control*. McGraw/Hill.

Bunn, D.W. (1975), “A Bayesian approach to the linear combination of forecasts”, *Operational ResearchQuarterly*, **26**, 325-329.

Bunn, D.W. (1981). Two methodologies for the linear combination of forecasts. *Journal of the OperationalResearch Society*,**32**, 213-222.

Broock, W. A., Scheinkman, J. A., Dechert, W. D. and LeBaron, B. (1996). A test for independence based on the correlation dimension. *Econometric Reviews*, **15(3)**, 197-235. doi: https://doi.org/10.1080/07474939608800353

Dickinson, J.P. (1973). Some statistical results on the combination of forecasts.*Operational ResearchQuarterly*,**24**, 253-260.

French, S. (1980). Outranking probabilities and the synthesis of forecasts.*Journal of the OperationalResearch Society*,**31**, 545-551.

Fiorucci, J. A., Pellegrini, T. R., Louzada, F., Petropoulos, F.and Koehler, A. B. (2016). Models for optimising the theta method and their relationship to state space models. *International Journal of Forecasting*, **32**(4), 1151–1161.

Gardner Jr, E. S. (1985). Exponential smoothing: The state of the art. *Journal of forecasting*, **4**(1), 1-28.

Genre, V, Kenny, G, Meyler, A and Timmermann, A (2013). Combining expert forecasts: Can anythingbeat the simple average?.*International Journal of Forecasting*,**29**(1), 108–121.

Granger, C. W. J.and Ramanathan, R. (1984). Improved methods of combining forecasts. *Journal ofForecasting*,**3**(2), 197–204.

Grushka-Cockayne, Y, Jose, VRR, and Lichtendahl Jr, KC (2017). Ensembles of Overfit and Overconfident Forecasts. *Management Science*, **63**(4), 1110–1130.

Hibon, M. and Evgeniou, T. (2005). To combine or not to combine: selecting among forecasts and their combinations. *International Journal of Forecasting*, **21**(1), 15–24.

Holt, C. E. (1957). *Forecasting seasonals and trends by exponentially weighted averages*. Carnegie Institute of Technology, Pittsburgh USA.

Hyndman, R. J., Koehler, A. B., Snyder, R. D. and Grose, S. (2002). A state space framework for automatic forecasting using exponential smoothing methods. *International Journal of forecasting*, **18**(3), 439-454.

Hyndman, R., Koehler, A. B., Ord, J. K. and Snyder, R. D. (2008). *Forecasting with exponential smoothing:the state space approach*. Springer-Verlag Berlin Heidelberg. ISBN: 978-3- 540-71918-2.

Hyndman, R. J. (2020). A brief history of forecasting competitions. *International Journal of Forecasting*, **36**(1), 7-14.

Jiang, P., Liu, Z., Niu, X.and Zhang, L. (2021). A combined forecasting system based on statistical method, artificial neural networks, and deep learning methods for short-term wind speed forecasting. *Energy*, 217, 119361.

Jose, V. R. R.and Winkler, R. L. (2008). Simple robust averages of forecasts: Some empirical results.*International Journal of Forecasting*,**24**(1), 163–169.

Jose, V. R. R., Grushka-Cockayne, Y. and Lichtendahl Jr, K. C. (2014). Trimmed opinion pools and the crowd's calibration problem. Management Science, **60**(2), 463-475.

Kourentzes, N., Barrow, D. K.and Crone, S. F. (2014). Neural network ensemble operators for time series forecasting. *Expert Systems with Applications*, **41**(9), 4235-4244.

Lichtendahl Jr, K. C. and Winkler, R. L. (2020). Why do some combinations perform better than others?.*International Journal of Forecasting*, **36**(1), 142-149.

Makridakis, S., Spiliotis, E.and Assimakopoulos, V. (2020). The M4 Competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, **36**(1), 54-74.

Pegels, C. C. (1969) Exponential Forecasting: Some New Variations. *Management Science*, **15**, 311-315.

Petropoulos, F.and Svetunkov, I. (2020). A simple combination of univariate models. *International journalof forecasting*,**36**(1), 110-115.

Reid, D.J. (1968).Combining three estimates of gross domestic product.*Economica*, **35**, 431-444.

Shaub, D. (2020). Fast and accurate yearly time series forecasting with forecast combinations. *International Journal of Forecasting*, **36**(1), 116-120.

Svetunkov, I. and Kourentzes, N. (2018). Complex exponential smoothing for seasonal time series. *WorkingPaper of Department of Management Science, Lancaster University*.**1**, 1–20.

Taylor, J. W. (2003). Exponential smoothing with a damped multiplicative trend. *International journal ofForecasting*,**19**(4), 715-725.

Taylor, J. W.and Taylor, K. S. (2023). Combining probabilistic forecasts of COVID-19 mortality in the United States. *European Journal of Operational Research*, **304**(1), 25-41.

Wang, X., Hyndman, R. J., Li, F. and Kang, Y. (2023). Forecast combinations: An over 50-year review. *International Journal of Forecasting*, **39**(4), 1518-1547.

Winters, P. R. (1960). Forecasting sales by exponentially weighted moving averages. *Management Science*, **6**(3), 324–342.