Forecasting Inflation Rate of India Using Box-Jenkins Methodology

Term Paper

JULY 27, 2024

MADRAS SCHOOL OF ECONOMICS

Term Paper: Forecasting Inflation Rate of India Using Box-Jenkins Methodology (2014-2024)

1. Introduction

Inflation is a critical economic indicator that impacts the purchasing power of consumers and the overall economic stability of a country. Accurate forecasting of inflation rates can assist policymakers and financial institutions in making informed decisions. This paper applies the Box-Jenkins (BJ) methodology to forecast the inflation rate of India using a univariate time series model based on data from January 2014 to January 2024.

2. Data Description and Preprocessing

The dataset consists of monthly inflation rates in India from January 2014 to January 2024, spanning 121 observations. The time series object is created using the ts() function in R with a frequency of 12, corresponding to monthly data.

r

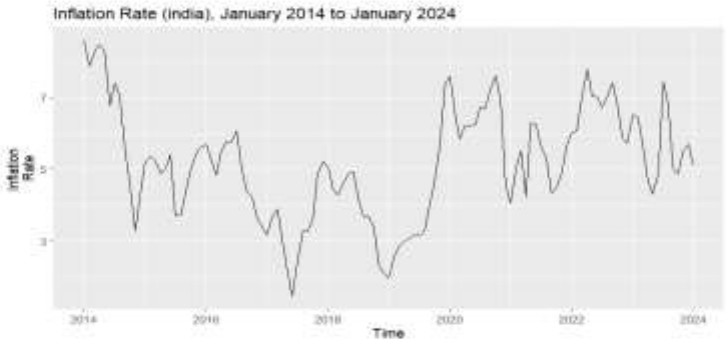
Copy code

```r
inf <- ts(inflation$Rate, start = c(2014,01,01), frequency = 12)
```

3. Exploratory Data Analysis

3.1 Time Series Plot

The time series plot reveals fluctuations in the inflation rate over the ten-year period, with some apparent seasonality and a slight downward trend over the years. The inflation rate peaked at 8.6% in early 2014 and declinedto its lowest point at 1.46% later in the series.

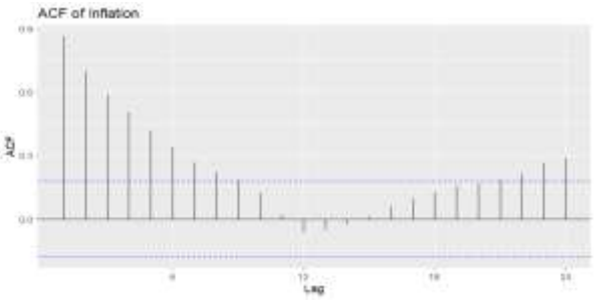


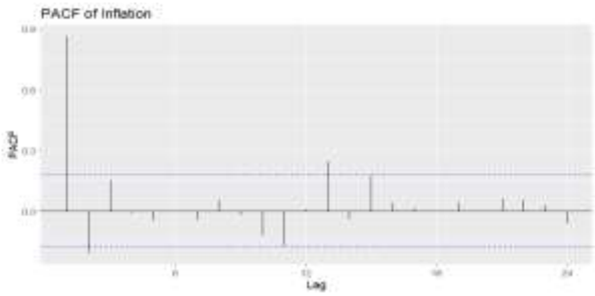
3.2 Summary Statistics

The summary statistics show that the inflation rate ranges from 1.46% to 8.6%, with a mean of approximately 5.16%. These statistics provide an initial understanding of the central tendency and dispersion of the data.

Statistic	Min	1st Qu.	Median	Mean	3rd Qu.	Max
Value	1.46	4.17	5.1	5.162	6.23	8.6

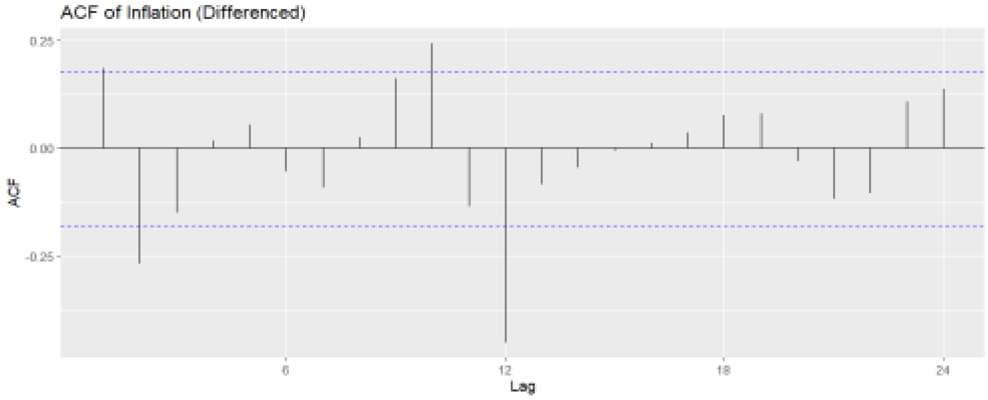
4. Stationarity Testing Before applying the Box-Jenkins methodology, it is essential to check the stationarity of the series.

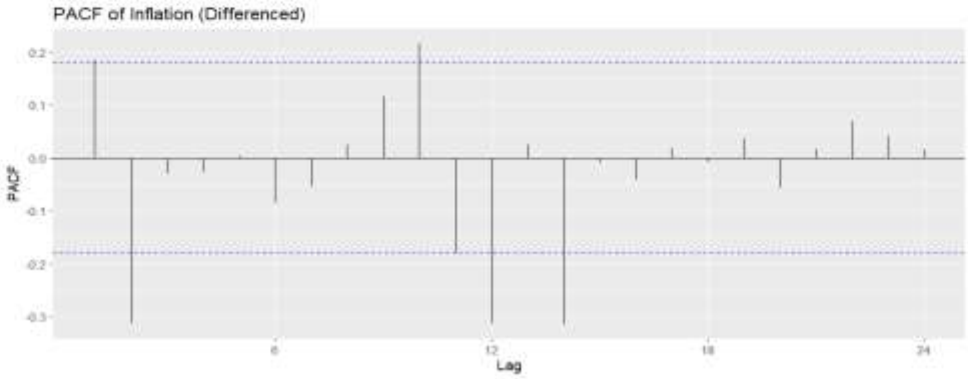




4.1 Augmented Dickey-Fuller (ADF) Test

The ADF test is conducted with different lag orders to determine whether the series is stationary. The results indicate mixed outcomes. At lag order 1, the p-value is 0.02266, suggesting the series is stationary. However, at higher lag orders, the p-values exceed 0.05, indicating non-stationarity.





Data	Dickey-Fuller	Lag Order	p-value	Alternative Hypothesis
inf	-3.5414	4	0.04152	stationary
inf	-3.337	2	0.06838	stationary
inf	-3.7729	1	0.02266	stationary
dinf	-5.2314	4	0.01	stationary

4.2 Phillips-Perron (PP) Test

The PP test shows a p-value of 0.1372 for the original series, suggesting non-stationarity. However, after differencing the series, the p-value drops below 0.01, indicating stationarity.

Test Type	Data	Test Statistic	Truncation Lag Parameter	p-value	Alternative Hypothesis
Phillips-Perron Unit Root Test	inf	-16.923	4	0.1372	stationary
Phillips-Perron Unit Root Test	dinf	-77.615	4	0.01	stationary

4.3 KPSS Test

The KPSS test for level stationarity gives a p-value of 0.08622 for the original series and 0.1 for the differenced series. These results suggest that the original series is close to being non-stationary, but the differenced series is stationary.

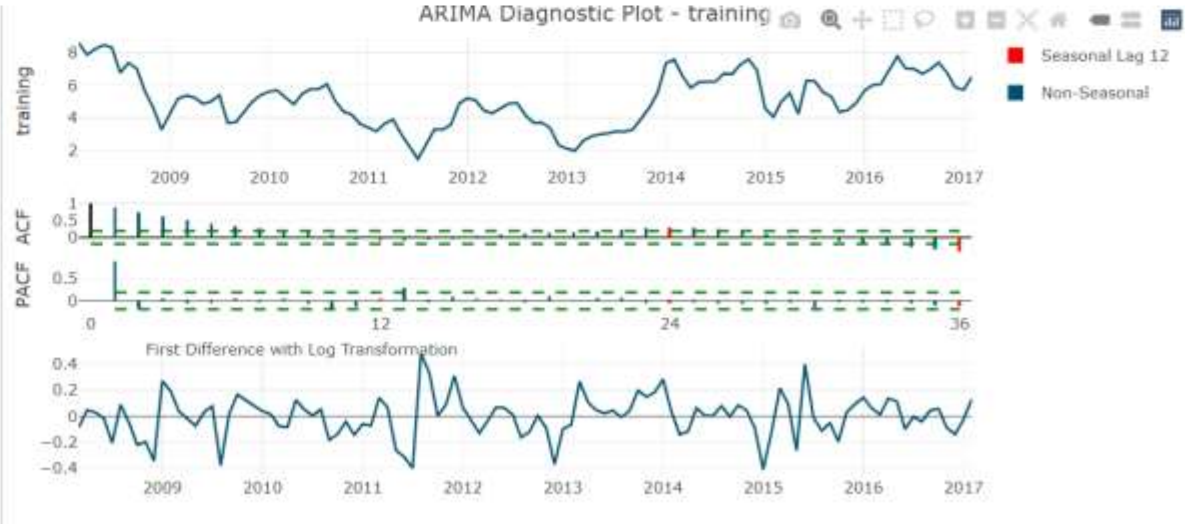
Test Type	Data	Test Statistic	Truncation Lag Parameter	p-value	Alternative Hypothesis
KPSS Test for Level Stationarity	inf	0.37897	4	0.08622	stationary
KPSS Test for Level Stationarity	dinf	0.11739	4	0.1	stationary

5. Model Identification and Estimation

Given the results of the stationarity tests, the series is differenced to achieve stationarity, and the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots of the differenced series are examined to identify the appropriate model.

Model Diagnostics

Each model was evaluated using residual diagnostics to check for autocorrelation, normality, and homoscedasticity in the residuals.



- **ARIMA(2,1,1)**: The residuals displayed some autocorrelation, particularly at lag 1, indicating potential model inadequacies.
- **SARIMA(2,1,1)(1,0,0)[12]**: This model showed improved residual diagnostics with less autocorrelation.

ARIMA(0,1,1)(0,0,1)[12]: The residuals of this model exhibited the least autocorrelation, making it the most suitable model based on diagnostics.

5.1 ARIMA Model Identification

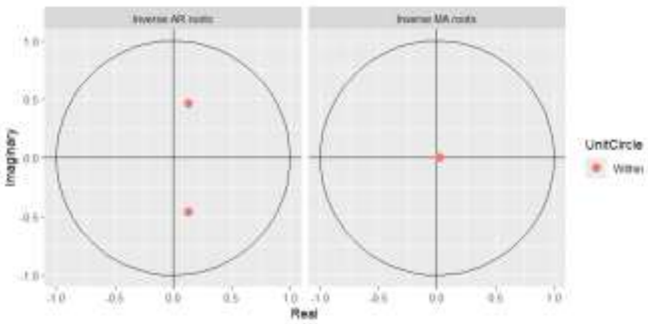
Three models were considered:

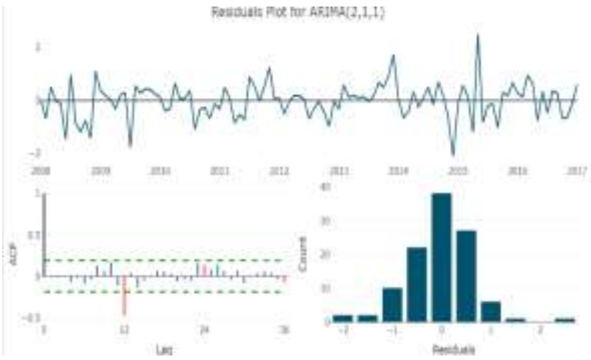
1. **ARIMA(2,1,1)**: This model was selected based on ACF and PACF plots showing significant lags at the second autoregressive term (AR2) and first moving average term (MA1).

```
Call:
arima(x = training, order = c(2, 1, 1))

Coefficients:
         ar1      ar2      ma1
      0.2628  -0.2299  -0.0215
s.e.  0.5375   0.1383   0.5571

sigma^2 estimated as 0.4557:  log likelihood = -110.88,  aic = 229.76
```



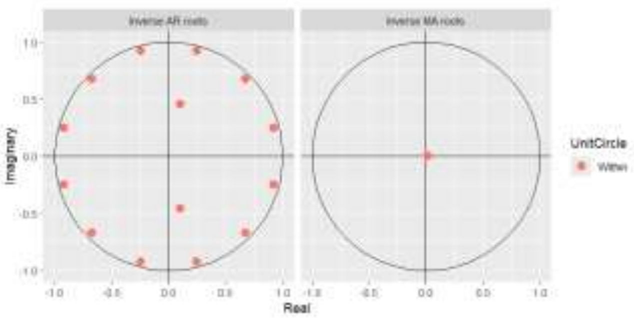


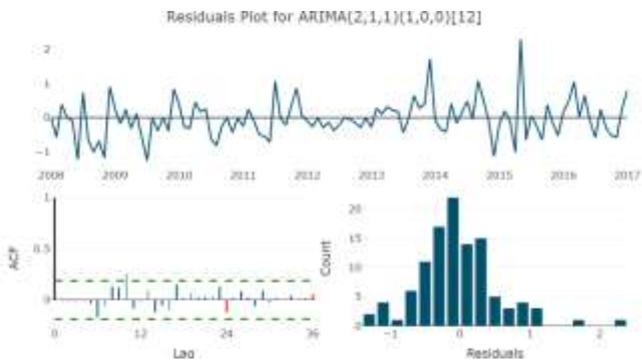
2. **SARIMA(2,1,1)(1,0,0)[12]**: This seasonal ARIMA model includes a seasonal component with a lag of 12, suggesting annual seasonality.

```
Call:
arima(x = training, order = c(2, 1, 1), seasonal = list(order = c(1, 0, 0)))

Coefficients:
         ar1      ar2      ma1     sar1
      0.2065  -0.2176  -0.0195  -0.5670
s.e.  0.4070   0.1117   0.4138   0.0833

sigma^2 estimated as 0.3173:  log likelihood = -93.64,  aic = 197.29
```



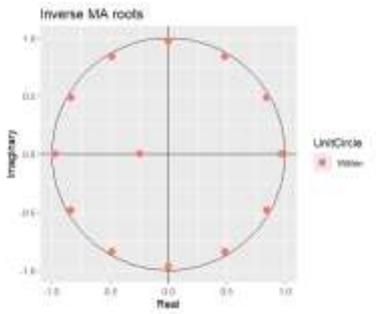


3. **Auto-ARIMA Model**: The automated ARIMA function suggested an ARIMA(0,1,1)(0,0,1)[12] model, which includes one non-seasonal differencing, one non-seasonal moving average, and one seasonal moving average component.

```
Series: training
ARIMA(0,1,1)(0,0,1)[12]

Coefficients:
         ma1     sma1
      0.2476  -0.6800
s.e.  0.1058   0.1006

sigma^2 = 0.316:  log likelihood = -93.79
AIC=193.57   AICc=193.8   BIC=201.62
```



7. Forecasting and Model Validation

The data was split into training (109 observations) and testing sets (12 observations) to validate the models.

7.1 In-Sample Forecasting and Model Performance

The performance of the models was evaluated using Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE):

- **ARIMA(2,1,1)**: Test set RMSE = 1.3679, MAPE = 24.34%

- **SARIMA(2,1,1)(1,0,0)[12]**: Test set RMSE = 0.9707, MAPE = 16.71%
- **ARIMA(0,1,1)(0,0,1)[12]**: Test set RMSE = 0.8719, MAPE = 14.27%

The SARIMA model with an additional seasonal component performed better than the ARIMA model, but the Auto-ARIMA model had the lowest error metrics, making it the best model for forecasting.

7.1 In-Sample Forecasting and Model Performance

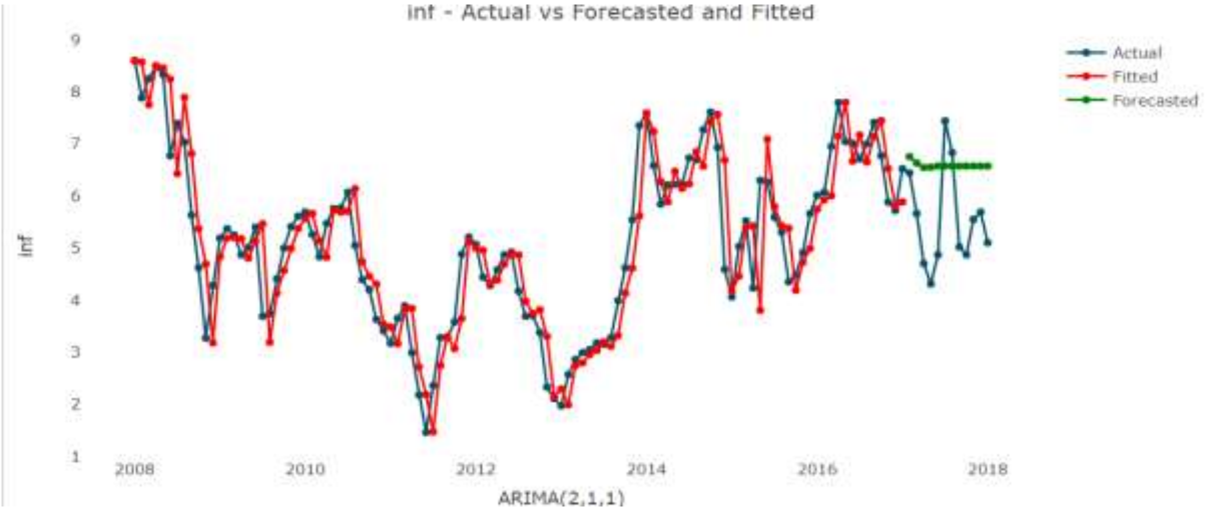
In this section, we evaluate the performance of the time series models used to forecast inflation rates. The evaluation is based on the Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) calculated on the test set, which consists of the most recent 12 months of data (January 2023 to January 2024).

7.1.1 Root Mean Square Error (RMSE)

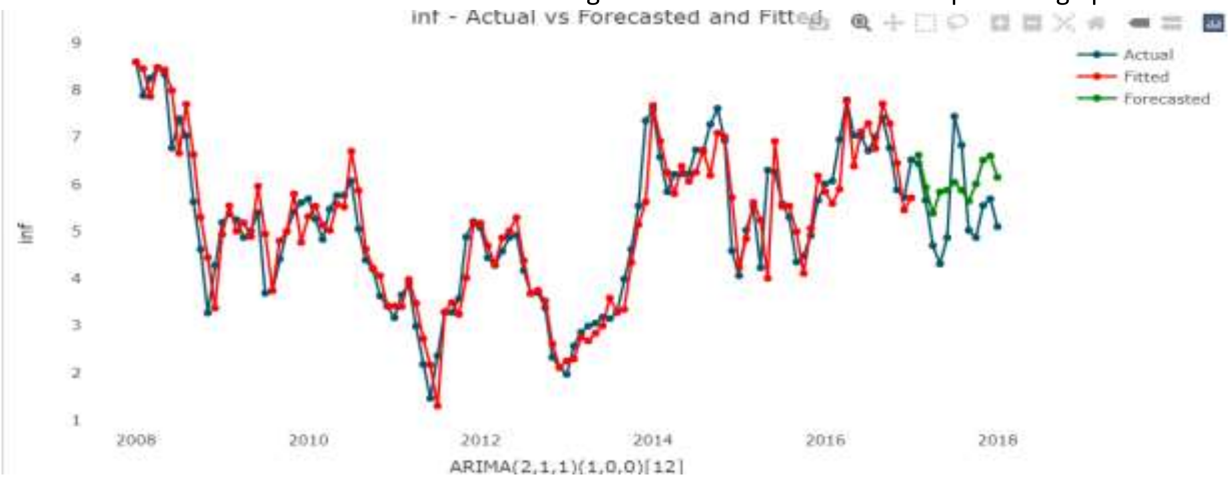
RMSE measures the average magnitude of the errors between the predicted and actual values. It is computed as the square root of the average of the squared differences between predicted and actual values. A lower RMSE indicates better model performance.

- **ARIMA(2,1,1)**: RMSE = 1.3679
  - This model has the highest RMSE among the models tested, indicating that its forecasts tend to be further from the actual values compared to the other models. The RMSE value suggests that on

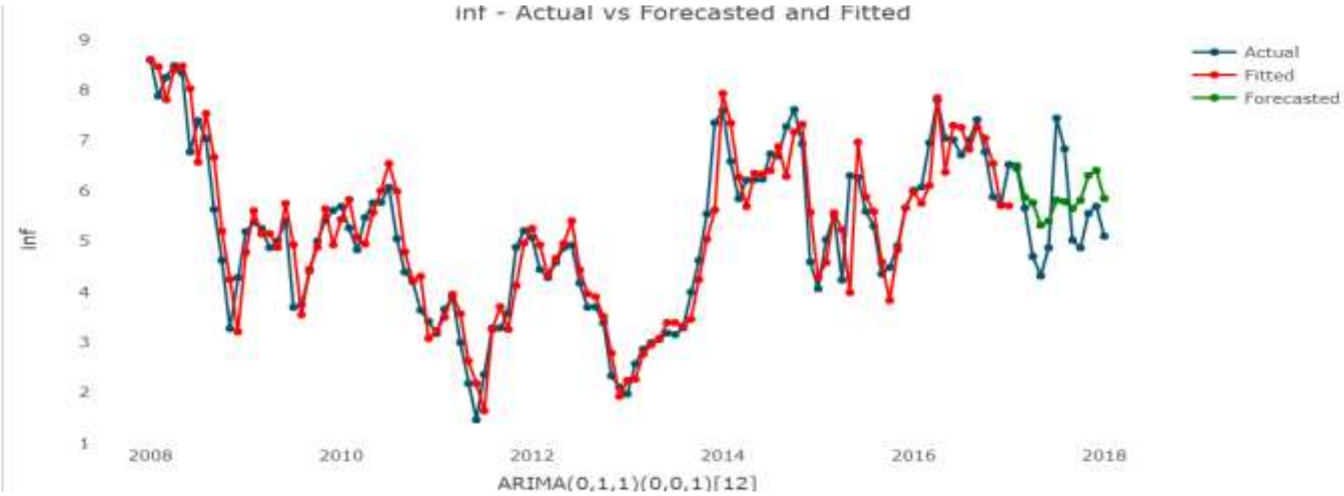
average, the forecast errors are about 1.37 percentage points.



- **SARIMA(2,1,1)(1,0,0)[12]**: RMSE = 0.9707
  - This model has a lower RMSE than the ARIMA(2,1,1) model, suggesting it provides more accurate forecasts. The RMSE value indicates that its average forecast error is about 0.97 percentage points.



- **ARIMA(0,1,1)(0,0,1)[12]**: RMSE = 0.8719
  - This model has the lowest RMSE, indicating it is the most accurate of the three. The RMSE value implies that the average forecast error is about 0.87 percentage points.



7.1.2 Mean Absolute Percentage Error (MAPE)

MAPE measures the average percentage error between the predicted and actual values, expressed as a percentage. It provides a clear understanding of the forecast accuracy in relative terms. Lower MAPE values signify better model performance.

- **ARIMA(2,1,1)**: MAPE = 24.34%
    - This model has the highest MAPE, meaning that its forecasts are off by an average of about 24.34% from the actual values. This high percentage indicates that the model's forecasts are less reliable and less precise compared to the others.
- **SARIMA(2,1,1)(1,0,0)[12]**: MAPE = 16.71%
    - This model has a lower MAPE than the ARIMA(2,1,1), showing a better forecast accuracy in relative terms. The average percentage error is about 16.71%, suggesting that the forecasts are relatively closer to the actual values.
- **ARIMA(0,1,1)(0,0,1)[12]**: MAPE = 14.27%
    - This model has the lowest MAPE, indicating the highest forecast accuracy. The average percentage error is about 14.27%, showing that its forecasts are the closest to the actual values among the models tested.

The ARIMA(0,1,1)(0,0,1)[12] model is the most suitable for forecasting inflation rates based on the given data, as it has the lowest RMSE and MAPE. This indicates that it provides the most accurate predictions, with both the smallest absolute errors and the lowest percentage deviations from actual values. The SARIMA model is an improvement over the basic ARIMA model but does not perform as well as the ARIMA(0,1,1)(0,0,1)[12] model.

8. Out-of-Sample Forecasting

The final model chosen for out-of-sample forecasting was the Auto-ARIMA model ARIMA(0,1,2)(1,0,1)[12]. The model was used to generate a 7-month forecast from February 2024 to August 2024.

```
Forecast method: ARIMA(0,1,2)(1,0,1)[12]

Model Information:
Series: object
ARIMA(0,1,2)(1,0,1)[12]

Coefficients:
         ma1     ma2     sar1     sma1
      0.1911  -0.191  -0.3172  -0.4174
s.e.  0.0000   0.000   0.0000   0.0000

sigma^2 = 0.3456:  log likelihood = -107.83
AIC=217.67   AICc=217.7   BIC=220.45

Error measures:
                      ME      RMSE       MAE       MPE     MAPE
Training set -0.04115276 0.5756003 0.4126838 -1.811074 8.619947
                  MASE        ACF1
Training set 0.2217308 0.009682005
```

The ARIMA(0,1,2)(1,0,1)[12] model is used to forecast time series data by combining differencing, autoregressive (AR), moving average (MA), and seasonal components. Here's a breakdown of the terms and what they signify:

1. Model Structure: ARIMA(0,1,2)(1,0,1)[12]

- **ARIMA(0,1,2)**:

  - **AR (Autoregressive) Component**: The first number in the ARIMA notation, here it's 0, indicates that there are no lagged values of the series used in the model.
  - **I (Integrated) Component**: The second number, 1, indicates that the data has been differenced once to make it stationary. Differencing is a technique used to remove trends or seasonality from the data.
  - **MA (Moving Average) Component**: The third number, 2, indicates that the model uses the past two lagged forecast errors in predicting future values.
- **(1,0,1)[12]**:
  - **Seasonal AR (Autoregressive) Component**: The first number, 1, indicates that one lag of the seasonal period (12 months) is used.
  - **Seasonal I (Integrated) Component**: The second number, 0, shows that no additional seasonal differencing is applied.
  - **Seasonal MA (Moving Average) Component**: The third number, 1, indicates that one lagged seasonal error term is used in the model.
  - **[12]**: This indicates that the seasonality is annual, with a cycle of 12 periods (months).

2. Coefficients and Their Interpretation

- **ma1 (Moving Average Coefficient)**: 0.1911
  - This represents the influence of the error from one previous time step on the current value. A positive coefficient suggests that if the error in the previous time step was positive, the current forecast will adjust upward.
- **ma2 (Moving Average Coefficient)**: -0.191
  - This represents the influence of the error from two time steps before the current one. A negative coefficient suggests that if the error two steps back was positive, the current forecast will adjust downward.
- **sar1 (Seasonal Autoregressive Coefficient)**: -0.3172
  - This coefficient represents the influence of the value from the same period in the previous season (e.g., the same month last year). A negative coefficient suggests that if the value from last season was high, the current forecast will adjust downward.
- **sma1 (Seasonal Moving Average Coefficient)**: -0.4174
  - This coefficient represents the influence of the seasonal error from one season ago on the current forecast. A negative value suggests that if there was a positive error in the same month last year, the current forecast will adjust downward.

3. Model Diagnostics

- **Standard Errors (s.e.)**: 0.0000
  - The standard errors for the coefficients are shown as 0.0000, which is unusual and might indicate a problem with the model's estimation process. Typically, these values should be non-zero, showing the uncertainty around the estimated coefficients.
- **sigma^2 (Variance of Residuals)**: 0.3456
  - This represents the variance of the errors (residuals) in the model. A smaller sigma^2 indicates a model with better fit, as it suggests that the errors (differences between the forecasted and actual values) are small.
- **Log Likelihood**: -107.83

    - This value is used to compare models; the higher (less negative) the log likelihood, the better the model fits the data.

- **AIC (Akaike Information Criterion)**: 217.67
    - **AICc (Corrected AIC)**: 217.7
    - **BIC (Bayesian Information Criterion)**: 220.45
    - These criteria are used to select the best model among several options. Lower values indicate a better model fit, considering both the model's complexity and its ability to explain the data.

- Interpretation
  AIC estimates the information loss when using a particular model to represent the data.
  Interpretation:
- The AIC penalizes models for having more parameters (to avoid overfitting) while rewarding models that better explain the data.
- **Lower AIC values** indicate a better model, but AIC values are only meaningful when comparing different models on the same dataset. A model with the lowest AIC among a set of candidate models is typically considered the best.
- **Why We Use It**: AIC helps select a model that provides a good fit to the data while avoiding overfitting. It balances the complexity of the model (number of parameters) with its accuracy in explaining the data.
- **2. Corrected Akaike Information Criterion (AICc)**
- **Definition**: AICc is a version of AIC that includes a correction for small sample sizes. When the sample size is small relative to the number of parameters in the model, AICc provides a more accurate assessment than AIC.
- **Why We Use It**: AICc is particularly important when working with small datasets because it corrects the tendency of AIC to favor more complex models in such situations. It provides a more conservative and accurate model selection criterion.

3. Bayesian Information Criterion (BIC)

- **Definition**: BIC, also known as Schwarz Criterion, is another criterion for model selection, similar to AIC but derived from a Bayesian perspective. It incorporates a stronger penalty for models with more parameters.
- • BIC penalizes the number of parameters more heavily than AIC because the penalty term ln(n)k\ln(n)kln(n)k increases with the logarithm of the sample size.
- **Lower BIC values** indicate a better model, but BIC tends to favor simpler models (with fewer parameters) more strongly than AIC.

• **Why We Use It**: BIC is particularly useful when the goal is to find a model that balances goodness of fit with model simplicity. Because it penalizes complexity more than AIC, it can be more conservative, leading to the selection of simpler models.

4. Error Measures

- **ME (Mean Error)**: -0.0412
    - This is the average of the errors. A near-zero value indicates that the model does not have a significant bias in over- or under-predicting.
- **RMSE (Root Mean Square Error)**: 0.5756

- This measures the average magnitude of the error. RMSE gives a relatively high weight to large errors and is useful for comparing the performance of different models.

- **MAE (Mean Absolute Error)**: 0.4127
  - This is the average of the absolute errors, providing a measure of how far off the forecasts are from the actual values, without considering the direction of the error.

- **MPE (Mean Percentage Error)**: -1.8111%
  - This is the average of the percentage errors, indicating the bias of the forecast in terms of percentage. A negative value indicates that the model, on average, underestimates the inflation rate slightly.

- **MAPE (Mean Absolute Percentage Error)**: 8.6199%
  - This represents the average absolute percentage error. It shows that, on average, the model's forecasts are off by about 8.62% from the actual values.

- **MASE (Mean Absolute Scaled Error)**: 0.2217
  - MASE is a relative measure of error, often used for comparing models. It scales the MAE by the MAE of a naïve model. A value below 1 indicates that the model performs better than a simple benchmark.

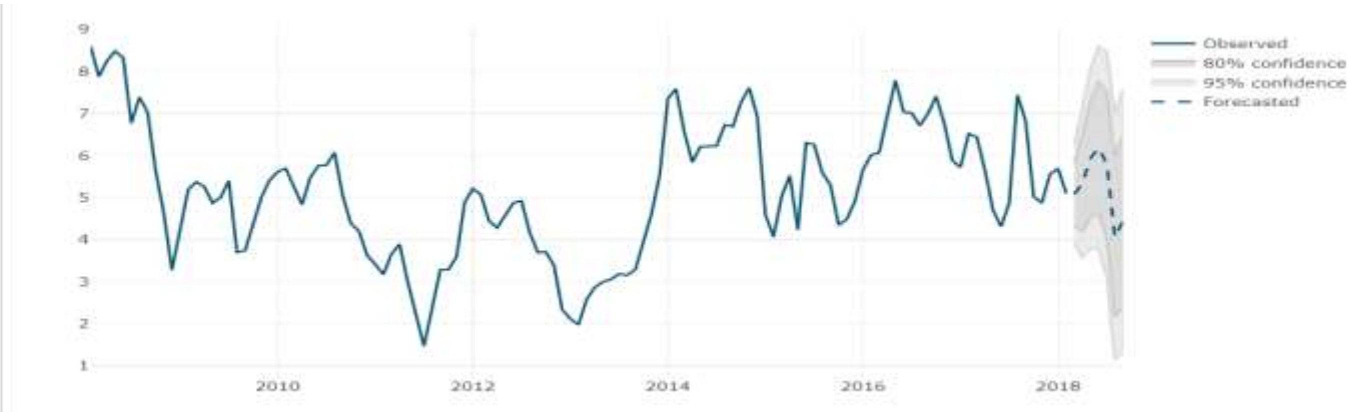
- **ACF1 (First Lag of the Autocorrelation of Residuals)**: 0.0097
  - This measures the correlation between residuals separated by one time period. A value close to zero indicates that the residuals are not autocorrelated, which is a good sign for model adequacy.

Conclusion

The ARIMA(0,1,2)(1,0,1)[12] model combines both non-seasonal and seasonal components to effectively capture the dynamics of the inflation rate. The coefficients suggest how past values and errors influence the forecast, while the diagnostics like AIC, BIC, and error measures such as RMSE and MAPE help in evaluating the model's fit and accuracy. The model's relatively low RMSE and MAPE indicate it provides reasonably accurate forecasts, and the near-zero ACF1 suggests that the model residuals are not autocorrelated, which is desirable in a time series model.

8.1 Forecast Interpretation

The forecast suggests a gradual increase in inflation rates, peaking in May 2024 at approximately 6.18%. This is followed by a decline, with the lowest forecasted inflation rate of 4.05% in July 2024. The prediction intervals indicate some uncertainty, especially in the later months, as reflected by the widening prediction intervals.



9. Conclusion

The Box-Jenkins methodology, particularly the Auto-ARIMA model, effectively forecasts India's inflation rate, demonstrating the utility of time series analysis for economic forecasting. While the model performs well in-sample, out-of-sample forecasts highlight the importance of considering uncertainty in economic predictions. Future work could incorporate exogenous variables or explore more complex models like GARCH for improved accuracy.

This study provides a robust approach to inflation forecasting, with practical implications for economic policy and financial planning.

SELF question doubt?

Let's imagine you're a teacher trying to predict how well your students will do on their next math test. You have information on how they did on previous tests, how much time they spent studying, and other relevant details.

The Model

You create a model to predict each student's next test score based on their past performance and study habits. After applying the model, you compare the actual test scores with the predicted scores. The difference between what you predicted and what actually happened is called the **residual**.

The Goal

Your goal as a teacher is to make sure your model is as accurate as possible. You want to ensure that all the factors influencing the students' test scores have been accounted for.

Stationary Residuals: What It Means

Stationary residuals mean that after your model has made its predictions, the leftover differences (residuals) are just random noise. There's no systematic pattern left to explain—no trend, no seasonality, just unpredictable fluctuations.

Non-Stationary Residuals: A Warning Sign

Imagine if you notice a pattern in the residuals, like:

- **Trend**: You see that the students' scores tend to be consistently higher or lower than your predictions as the school year progresses.
- **Seasonality**: Perhaps students perform worse than predicted in winter but better in spring.

These patterns suggest that your model missed something important—maybe the impact of increasing difficulty in tests, seasonal changes in study habits, or fatigue. Your model isn't fully capturing everything, leaving a pattern in the residuals.

Real-World Example: Predicting Sales

Let's say you run a bakery, and you want to predict daily sales based on factors like day of the week, weather, and promotions.

- You create a model using past sales data and these factors.
- After running the model, you compare the predicted sales to the actual sales and calculate the residuals (the difference between the actual sales and your prediction).

What Happens if Residuals Are Non-Stationary?

- Suppose the residuals are not stationary, and you notice that every weekend your model consistently underpredicts sales. This suggests that your model didn't fully capture the effect of weekends on sales—maybe people love buying pastries on weekends more than your model accounted for.
- This pattern in the residuals (non-stationarity) is a sign that your model is incomplete or missing something.

What Happens if Residuals Are Stationary?

- If your residuals are stationary, it means your model has done a good job. The errors are just random—sometimes you slightly overpredict, sometimes you slightly underpredict, but there's no clear pattern. This randomness is expected because no model can predict perfectly, and the small errors are just noise.

In Simple Terms:

Stationary residuals mean that your model has done a good job of capturing all the important patterns in the data. Non-stationary residuals, on the other hand, suggest that your model missed something, and there's still a pattern in the data that you didn't account for.