

# Methods Explanation

January 25, 2026

The key objective is to compare several forecasting strategies under (i) a single mean break and (ii) multiple mean breaks, and determine which method is most accurate according to standard forecast evaluation metrics.

## 1 Data Generating Processes (DGPs)

We simulate autoregressive time series with deterministic mean shifts. The baseline dynamics follow an AR(1) process, but the mean changes at one or more break dates.

### 1.1 Single-break DGP

The single-break process is:

$$y_t = \mu_t + \phi y_{t-1} + \varepsilon_t \text{ seenote}, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2), \quad (1)$$

where the mean evolves as:

$$\mu_t = \begin{cases} \mu_0 & \text{if } t \leq T_b, \\ \mu_1 & \text{if } t > T_b. \end{cases} \quad (2)$$

Hence, there are two regimes (pre-break and post-break) and one break date  $T_b$ .

### 1.2 Multiple-break DGP

The multiple-break process extends the mean to three regimes with two break dates  $b_1$  and  $b_2$ :

$$y_t = \mu_t + \phi y_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2), \quad (3)$$

with:

$$\mu_t = \begin{cases} \mu_0 & \text{if } t \leq b_1, \\ \mu_1 & \text{if } b_1 < t \leq b_2, \\ \mu_2 & \text{if } t > b_2. \end{cases} \quad (4)$$

This design produces multiple structural shifts in the mean.

## 2 Forecasting Methods

For each simulated series, we compute a one-step-ahead forecast after the break (single case) or after the last break (multiple case). Let  $y_{t_0}$  be the true next observation and  $\hat{y}_{t_0}$  be the forecast. The forecast error is:

$$e = y_{t_0} - \hat{y}_{t_0}. \quad (5)$$

## 2.1 Global AR(1) (ARIMA(1,0,0))

The global approach estimates one AR(1) model using all available training data:

$$y_t = c + \phi y_{t-1} + u_t. \quad (6)$$

The one-step forecast is:

$$\hat{y}_{t+1} = \hat{c} + \hat{\phi} y_t. \quad (7)$$

**Why it can perform poorly under breaks:** the model pools observations from different mean regimes, so  $\hat{c}$  may become a compromise between regimes, leading to biased forecasts after structural breaks.

## 2.2 Rolling AR(1) (ARIMA(1,0,0) on a moving window)

The rolling method estimates the same AR(1) specification but only using the most recent  $w$  observations:

$$\{y_{t-w+1}, \dots, y_t\}. \quad (8)$$

**Why it helps:** after a break, recent observations contain more post-break information, allowing faster adaptation to the new mean. **Trade-off:** smaller samples can increase estimation noise.

## 2.3 AR(1) + Break Dummy (Oracle)

This method explicitly models breaks via dummy variables. It is called **oracle** because it assumes the break date(s) are known (true in simulation, usually unknown in real data).

### 2.3.1 Single-break dummy model

Define:

$$d_t = \mathbf{1}(t > T_b), \quad (9)$$

and estimate:

$$y_t = c + \phi y_{t-1} + \delta d_t + u_t. \quad (10)$$

The one-step forecast becomes:

$$\hat{y}_{t+1} = \hat{c} + \hat{\phi} y_t + \hat{\delta} d_{t+1}. \quad (11)$$

**Interpretation:** the intercept shifts by  $\delta$  after the break, so the model can match the mean change.

### 2.3.2 Multiple-break dummy model

Define two dummies:

$$d_{1,t} = \mathbf{1}(t > b_1), \quad d_{2,t} = \mathbf{1}(t > b_2), \quad (12)$$

and estimate:

$$y_t = c + \phi y_{t-1} + \delta_1 d_{1,t} + \delta_2 d_{2,t} + u_t. \quad (13)$$

Forecast:

$$\hat{y}_{t+1} = \hat{c} + \hat{\phi} y_t + \hat{\delta}_1 d_{1,t+1} + \hat{\delta}_2 d_{2,t+1}. \quad (14)$$

**Why it is strong:** the model adjusts the mean at each break, matching the DGP structure.

## 2.4 AR(1) + Estimated Break (Grid Search) — Single Break

In practice, break dates are often unknown. This method estimates a single break date  $\hat{T}_b$  by searching over candidate break points and choosing the one that minimizes the sum of squared errors (SSE) from two segment-specific AR(1) fits:

$$\hat{T}_b = \arg \min_{T_b} (SSE_1(T_b) + SSE_2(T_b)). \quad (15)$$

After estimating  $\hat{T}_b$ , forecasting is performed using the most recent regime (post-break segment). **Why it can be worse than oracle dummies:** estimation error in  $\hat{T}_b$  can lead to incorrect segmentation and larger forecast errors.

## 2.5 Markov Switching (Two Regimes)

Markov switching models allow regime changes without specifying break dates. A simplified interpretation is:

$$y_t = c_{s_t} + \phi y_{t-1} + u_t, \quad (16)$$

where  $s_t \in \{1, 2\}$  is an unobserved regime evolving according to Markov transition probabilities:

$$P(s_t = j \mid s_{t-1} = i) = p_{ij}. \quad (17)$$

**Practical issue:** estimation relies on non-linear maximum likelihood and can be numerically fragile; convergence failures may occur frequently in automated Monte Carlo loops.

## 3 Monte Carlo Design

For each scenario (single or multiple breaks), the simulation proceeds as follows:

1. Generate a time series  $\{y_t\}_{t=1}^T$  from the relevant DGP.
2. Choose the forecast time after the break(s):
  - Single break:  $t_0 = T_b + 20$ .
  - Multiple breaks:  $t_0 = b_2 + 20$ .
3. Fit each forecasting method using training data  $\{y_1, \dots, y_{t_0-1}\}$  and compute a one-step forecast  $\hat{y}_{t_0}$ .
4. Compute forecast error  $e = y_{t_0} - \hat{y}_{t_0}$ .
5. Repeat steps (1)–(4) for  $N$  replications (e.g.  $N = 200$ ) to obtain an empirical distribution of errors for each method.

## 4 Evaluation Metrics

Let  $\{e_i\}_{i=1}^N$  be the forecast errors for a method across Monte Carlo replications. We compute:

## 4.1 Bias

$$\text{Bias} = \frac{1}{N} \sum_{i=1}^N e_i. \quad (18)$$

Bias indicates systematic under- or over-forecasting. Values close to zero imply the method is not systematically wrong.

## 4.2 Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |e_i|. \quad (19)$$

MAE measures the typical magnitude of forecast errors.

## 4.3 Root Mean Squared Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N e_i^2}. \quad (20)$$

RMSE penalizes large errors more heavily than MAE. In this study, we rank methods primarily by RMSE (lower is better).

# 5 Results and Interpretation

The comparison table reports RMSE, MAE, Bias, and the number of successful replications  $N$  (and failure counts where relevant).

## 5.1 Single-break case

In the single-break scenario, the ranking by RMSE typically shows:

- **AR(1) + Break Dummy (oracle)** performs best (lowest RMSE/MAE, bias near zero).
- **AR(1) + Estimated Break (grid)** usually performs worse than oracle dummies due to break date estimation error.
- **Rolling AR(1)** improves over global AR(1) by adapting to recent data, but remains inferior to explicit break modeling.
- **Global AR(1)** performs worst because it mixes pre- and post-break regimes, generating biased forecasts after the break.

## 5.2 Multiple-break case

In the multiple-break scenario:

- **AR(1) + Break Dummy (oracle)** again performs best, because multiple dummies adjust the mean at each break.
- **Rolling AR(1)** typically performs better than global AR(1) because it focuses on recent data dominated by the final regime.
- **Global AR(1)** remains weaker as it pools observations from all regimes, increasing bias and error.

## 5.3 Markov Switching results

In the reported output, Markov switching produced NaN metrics with  $N = 0$  and high failure counts. This indicates that the model did not successfully converge or produce forecasts during the Monte Carlo loop in the implemented settings. This is consistent with the known numerical sensitivity of regime-switching maximum likelihood estimation. Therefore, Markov switching performance is not reported in those runs.

# 6 Overall Conclusion

Across both single and multiple break designs, the strongest forecasting performance is achieved by the **AR(1) + Break Dummy (oracle)** specification, as it explicitly incorporates the structural changes in the mean and therefore produces the lowest RMSE and MAE and bias close to zero.

However, it is important to emphasize that the break dummy approach is an **oracle benchmark** because it assumes the break date(s) are known. In real applications where break dates are unknown, adaptive approaches such as rolling estimation or break detection procedures become necessary, though they may produce slightly higher forecast errors than the oracle benchmark.