Structural Break Forecasting: A Monte Carlo Study

Research Module in Econometrics and Statistics
Fundamentals of Monte Carlo Simulations in Data Science

Aadya Khatavkar
University of Bonn
s38akhat@uni-bonn.de

Winter Semester 2025/26

Abstract

This study evaluates forecasting performance under **structural breaks** using Monte Carlo simulations. We implement data-generating processes for three break types: variance breaks (volatility shifts), mean breaks (intercept shifts), and parameter breaks (AR coefficient shifts). Forecasting methods include global ARIMA, rolling-window ARIMA with automatic order selection, GARCH, post-break estimation, and Markov switching models. We extend the analysis to heavy-tailed (Student-t) distributions and implement optimal window selection via grid search following Pesaran (2013). Evaluation uses both point forecast metrics (RMSE, MAE, Bias) and uncertainty quantification (Coverage, Log-score). Results inform practical guidance on method selection under parameter instability.

Keywords: Structural breaks, Monte Carlo simulation, ARIMA, GARCH, rolling window, variance breaks, heavy tails

Contents

1 Introduction

Time series forecasting faces a fundamental challenge when the data-generating process undergoes *structural breaks*—discrete parameter changes at specific points in time. Such breaks are pervasive in economic and financial data, arising from policy changes, financial crises, and regime shifts.

This project develops a comprehensive Monte Carlo framework to evaluate forecasting methods under three types of structural breaks:

1. **Variance breaks:** Shifts in innovation volatility ($\sigma_1^2 \to \sigma_2^2$)
2. **Mean breaks:** Shifts in the intercept ($\mu_0 \to \mu_1$)
3. **Parameter breaks:** Shifts in the AR coefficient ($\phi_1 \to \phi_2$)

The research questions are:

1. How do different forecasting methods perform under each break type?
2. What is the optimal rolling window size for different break magnitudes?
3. How do heavy-tailed distributions affect forecasting accuracy?
4. Can adaptive methods match oracle specifications that know break dates?

2 Data-Generating Processes

All DGPs are based on AR(1) processes with structural breaks at known points.

2.1 Variance Break DGP

The variance-break DGP is:

$$y_t = \mu + \phi y_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma_t^2), \tag{1}$$

where:

$$\sigma_t^2 = \begin{cases} \sigma_1^2 & \text{if } t \leq T_b, \\ \sigma_2^2 & \text{if } t > T_b. \end{cases} \tag{2}$$

Parameters: $T = 400$, $T_b = 200$, $\phi = 0.6$, $\sigma_1 = 1.0$, $\sigma_2/\sigma_1 \in \{1.5, 2.0, 3.0, 5.0\}$.

2.2 Mean Break DGP

The mean-break DGP is:

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

where:

$$\mu_t = \begin{cases} \mu_0 & \text{if } t \leq T_b, \\ \mu_1 & \text{if } t > T_b. \end{cases} \tag{4}$$

Parameters: $T = 300$, $T_b = 150$, $\phi = 0.6$, $\mu_0 = 0$, $\mu_1 = 2$.

2.3 Parameter Break DGP

The parameter-break DGP is:

$$y_t = \phi_t y_{t-1} + \varepsilon_t, \quad \varepsilon_t \overset{\text{iid}}{\sim} N(0, \sigma^2), \tag{5}$$

where:

$$\phi_t = \begin{cases} \phi_1 & \text{if } t \leq T_b, \\ \phi_2 & \text{if } t > T_b. \end{cases} \tag{6}$$

Parameters: $T = 400$, $T_b = 200$, $\phi_1 = 0.2$, $\phi_2 = 0.9$.

2.4 Heavy-Tailed Innovations

For robustness, we consider Student-t distributed innovations:

$$\varepsilon_t = \frac{z_t}{\sqrt{\nu/(\nu-2)}}, \quad z_t \sim t_\nu. \tag{7}$$

The denominator $\sqrt{\nu/(\nu-2)}$ standardizes to unit variance, enabling fair comparison with Gaussian innovations. Default: $\nu = 3$ (heavy tails).

3 Forecasting Methods

We implement six forecasting approaches.

3.1 Global ARIMA

Fits ARIMA(p, d, q) on **all training data**. Order is either fixed or auto-selected via AIC:

$$(p^*, d^*, q^*) = \arg\min_{p,d,q} \text{AIC}(p, d, q). \tag{8}$$

Limitation: After breaks, estimates are contaminated by pre-break data.

3.2 Rolling-Window ARIMA

Estimates ARIMA using only the most recent w observations:

$$\text{Training: } \{y_{t-w+1}, \ldots, y_t\}. \tag{9}$$

Key trade-off (Pesaran 2013):

- Small w: Fast adaptation, high variance
- Large w: Low variance, includes pre-break data

Optimal w depends on break magnitude—larger breaks favor smaller windows.

3.3 GARCH(1,1)

Models conditional variance directly:

$$y_t = \mu + \varepsilon_t, \quad \varepsilon_t = \sigma_t z_t, \tag{10}$$

$$\sigma_t^2 = \omega + \alpha\varepsilon_{t-1}^2 + \beta\sigma_{t-1}^2. \tag{11}$$

Strength: Naturally adapts to volatility changes via α and β.

3.4 Post-Break ARIMA

1. Estimate break point $\hat{T}_b$ via SSE minimization
2. Fit ARIMA on post-break data: $\{y_{\hat{T}_b+1}, \ldots, y_t\}$

Falls back to global if insufficient post-break observations.

3.5 Averaged Window

Averages forecasts across multiple window sizes to reduce sensitivity to window choice:

$$\hat{y}_{t+h} = \frac{1}{K}\sum_{k=1}^{K} \hat{y}_{t+h}^{(w_k)}. \tag{12}$$

3.6 Markov Switching

Regime-switching model with unobserved state $s_t \in \{1, 2\}$:

$$y_t = c_{s_t} + \phi y_{t-1} + \varepsilon_t, \quad P(s_t = j | s_{t-1} = i) = p_{ij}. \tag{13}$$

Caveat: Numerically sensitive; convergence failures common in MC loops.

4 Monte Carlo Design

4.1 Simulation Procedure

For each scenario:

1. Generate $\{y_t\}_{t=1}^T$ from relevant DGP
2. Split: training $\{y_1, \ldots, y_{T-h}\}$, test $\{y_{T-h+1}, \ldots, y_T\}$
3. Fit each method on training data
4. Compute h-step forecasts and intervals
5. Evaluate metrics
6. Repeat for $N = 200$ replications

4.2 Grid Search for Optimal Window

Following Pesaran (2013), we search over:

- Window sizes: $w \in \{20, 50, 100, 150, 200\}$
- Break magnitudes: $\sigma_2/\sigma_1 \in \{1.5, 2.0, 3.0, 5.0\}$

This produces a **loss surface** for optimal window selection.

Practitioner note: Grid search informs fixed-window policies but should not be applied adaptively in real-time (look-ahead bias).

5 Evaluation Metrics

5.1 Point Forecast Metrics

Let $e_i = y_i - \hat{y}_i$ denote forecast errors.

RMSE:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} e_i^2} \tag{14}$$

MAE:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |e_i| \tag{15}$$

Bias:

$$\text{Bias} = \frac{1}{N}\sum_{i=1}^{N} e_i \tag{16}$$

5.2 Uncertainty Metrics

Interval Coverage:

$$\text{Coverage}_\alpha = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left(y_i \in \text{CI}_\alpha\right) \tag{17}$$

Log-Score (proper scoring rule):

$$\text{LogScore} = \frac{1}{N}\sum_{i=1}^{N}\left[-\frac{1}{2}\log(2\pi\hat{\sigma}_i^2) - \frac{e_i^2}{2\hat{\sigma}_i^2}\right] \tag{18}$$

6 Implementation Summary

6.1 Code Organization

Table 1: Module Overview

Module	Location	Key Functions
DGPs	`dgps/static.py`	`simulate_variance_break()`, etc.
Forecasters	`estimators/forecasters.py`	ARIMA, GARCH, Markov
MC Engine	`analyses/simulations.py`	`mc_variance_breaks()`
Visualization	`analyses/plots.py`	Loss surfaces, comparisons

6.2 Key Technical Features

1. **Automatic ARIMA order selection** via AIC/BIC grid search
2. **Heavy-tailed distributions** with standardized Student-t innovations
3. **Unified simulation engine** handling all break types
4. **Realized volatility functions** for empirical applications
5. **Scenario-based configuration** via JSON files

7 Results

Table 2: Variance Break: Method Comparison (Placeholder)

Method	RMSE	MAE	Bias	Cov80	Cov95	LogScore
ARIMA Global	—	—	—	—	—	—
ARIMA Rolling	—	—	—	—	—	—
GARCH(1,1)	—	—	—	—	—	—
ARIMA Post-Break	—	—	—	—	—	—

Table 3: Loss Surface: RMSE by Window and Break Magnitude (Placeholder)

Window	$\sigma_2 = 1.5\sigma_1$	$\sigma_2 = 2\sigma_1$	$\sigma_2 = 3\sigma_1$	$\sigma_2 = 5\sigma_1$
$w = 20$	—	—	—	—
$w = 50$	—	—	—	—
$w = 100$	—	—	—	—
$w = 200$	—	—	—	—

8 Conclusion

This project provides a comprehensive Monte Carlo framework for evaluating forecasting under structural breaks. Key contributions:

1. **Unified DGP framework** for variance, mean, and parameter breaks
2. **Multiple forecasting methods** from simple ARIMA to Markov switching
3. **Heavy-tailed extensions** for realistic financial data
4. **Optimal window selection** via Pesaran (2013) grid search
5. **Both point and probabilistic** evaluation metrics

Practical implications:

- GARCH adapts well to variance breaks
- Rolling windows outperform global estimation under breaks
- Optimal window decreases with break magnitude
- Heavy tails require larger samples for stable estimation

8.1 Future Work

- S&P 500 realized volatility application (Thomson Reuters Eikon)
- Multi-step ahead forecasting
- ARIMA + GARCH ensemble methods
- Online break detection and adaptive windowing

References

Box, G. E. and Jenkins, G. M. (1970). *Time Series Analysis: Forecasting and Control.* Holden-Day.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica*, 50(4):987–1007.

Francq, C. and Zakoïan, J. M. (2019). *GARCH Models.* Wiley, 2nd edition.

Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series. *Econometrica*, 57(2):357–384.

Pesaran, M. H. (2013). The role of structural breaks in forecasting. In *Handbook of Economic Forecasting*, volume 2B, pages 1159–1191. Elsevier.

Bai, J. and Perron, P. (1998). Estimating and testing linear models with multiple structural changes. *Econometrica*, 66(1):47–78.