IPRPD
International Journal of Business & Management Studies
ISSN 2694-1430 (Print), 2694-1449 (Online)
Volume 07; Issue no 04: April, 2026
DOI: 10.56734/ijbms.v7n4a3

CAN ADAPTIVE MEMORY SYSTEMS OUTPERFORM HAR? EVIDENCE FROM REALIZED VOLATILITY FORECASTING

Yuan Wen[1]

[1]*State University of New York at New Paltz*

Abstract

We compare the Heterogeneous Autoregressive (HAR) model with a novel Continuous Memory System (CMS) for forecasting realized volatility. CMS employs 12 exponential moving averages with adaptive decay rates modulated by learned, level-specific shock sensitivities through a rank-1 gating mechanism. The response of each memory level to volatility shocks is governed by an optimized sensitivity parameter that determines whether the level accelerates or decelerates during turbulent periods. Using 1,234 daily observations from February 2021 to January 2026, we estimate the model through bounded constrained optimization and compare its performance with that of the parsimonious HAR benchmark.

CMS learns a surprisingly intuitive pattern in how different time horizons respond when markets become turbulent. Short-term memory reacts aggressively to volatility spikes, updating rapidly to capture sudden regime shifts. Medium- to long-term memory behaves in the opposite way, slowing down sharply during stress to preserve a stable baseline, with the strongest dampening occurring at horizons of roughly 4 to 6 weeks (levels 10 and 11). This creates an asymmetric response pattern: high reactivity at short horizons and strong stabilization at medium to long horizons. Notably, the model discovers this structure automatically from the data, without being explicitly designed to behave in this way, and the resulting pattern aligns closely with financial intuition about how different forecast horizons should weight past information during volatile periods. The sole exception is the 60-day horizon (level 12), which exhibits a large positive sensitivity. This may reflect distinct very long-term dynamics, or it may be an overfitting artifact, so it should be interpreted with caution.

Despite its theoretical appeal, CMS underperforms in practice. Its out-of-sample forecasting error is 30% higher than that of HAR, even though it fits the training data extremely well. This is consistent with the classic problem of overfitting, in which a model captures historical patterns too closely and then fails to generalize well to new observations. The added complexity of CMS, with 25 tunable parameters versus HAR's 4, appears to be a liability rather than an asset in a limited-sample setting. The sophisticated level-specific shock responses also provide almost no forecasting improvement, only 0.13%, over a simpler uniform-gating specification, while also creating numerical instability at very short horizons. Ultimately, HAR's simplicity is a strength: fewer parameters leave less room for overfitting, making the model more reliable out of sample. At the same time, the response patterns learned by CMS, particularly how different forecast horizons adjust to volatility shocks, provide useful economic intuition that may help guide the design of better hybrid models in future work.

Keywords

Realized Volatility, HAR Model, Adaptive Filtering, Machine Learning, Gating Mechanisms

1. Introduction

The forecasting of realized volatility has been dominated by the Heterogeneous Autoregressive (HAR) model of Corsi (2009), which captures volatility clustering through simple averages at daily, weekly, and monthly horizons. Despite its parsimony, HAR has proven remarkably robust across asset classes, time periods, and market regimes. Its success stems from recognizing that volatility exhibits components operating on different time scales, a stylized fact well-documented in the financial econometrics literature (Andersen et al., 2003; Müller et al., 1997).

Despite its empirical success, the HAR model relies on manually specified aggregation windows and fixed linear dynamics, implicitly assuming that the relevance of each time horizon remains stable over time. In practice, however, volatility dynamics are nonstationary: episodes of financial stress, macroeconomic transitions, and changes in market microstructure can alter both volatility persistence and the dominant temporal scale. These limitations have motivated a range of extensions, such as time-varying parameters, regime switching, and nonlinear dynamics, but most retain a reliance on predefined horizons and static memory structures.

Recent advances in machine learning suggest potential gains from adaptive models that dynamically adjust their memory structures in response to market conditions. Rather than fixed horizons, such models could extend or contract their lookback windows based on volatility regime transitions, potentially capturing regime changes more effectively than static specifications.

This paper introduces the Continuous Memory System (CMS), an adaptive forecasting framework that extends exponential moving averages (EMAs) with level-specific gating mechanisms. Unlike HAR's fixed boxcar horizons, CMS maintains 12 memory levels with baseline half-lives spanning 1 to 60 days. Each level's decay rate is modulated by a learned sensitivity parameter that determines how strongly that horizon responds to volatility shocks. The gating mechanism follows a rank-1 structure: a common surprise intensity signal (based on recent volatility changes) interacts multiplicatively with level-specific sensitivities to produce time-varying effective decay rates.

A concrete formulation of this framework is provided by Behrouz et al. (2025), who introduce *nested learning* as a general paradigm for modeling learning systems with implicit long-term memory. Nested learning represents a model and its training dynamics as a hierarchy of interacting optimization processes operating at different temporal frequencies, allowing multi-timescale behavior to emerge endogenously rather than being imposed through predefined aggregation windows. Within this framework, the authors propose the CMS as a concrete architectural instantiation of nested learning: a cascaded system of memory blocks that update at different rates and thereby store information at different effective time scales. CMS generalizes the traditional short-term/long-term memory dichotomy by allowing memory persistence to arise from the update schedule itself. While nested learning and CMS were developed in a general learning context, their structure closely parallels the heterogeneous-horizon intuition underlying econometric volatility models such as HAR.

In this paper, we adapt the CMS architecture to a volatility forecasting setting and compare its empirical performance with that of the HAR model using U.S. equity volatility data. Whereas HAR encodes heterogeneous horizons through fixed rolling averages, CMS implements adaptive memory states whose effective time scales evolve over time. Our primary research questions are: (1) Can an adaptive memory system learn economically meaningful shock responses? (2) Does adaptation improve out-of-sample forecasting performance relative to the parsimonious HAR specification?

We find nuanced answers: CMS successfully learns an interpretable adaptive structure. Short-term horizons (1–1.5 days, levels 1–2) exhibit high positive shock sensitivity, amplifying decay rates during volatility spikes to track new regimes immediately. Intermediate horizons (2–9 days, levels 3–7) show near-zero mixed sensitivity, remaining largely neutral to shocks. Medium-to-long horizons (14–41 days, levels 8–11) exhibit negative sensitivity — increasingly so at longer scales, from moderate dampening at levels 8–9 (14–20 days) to strong dampening at levels 10–11 (29–41 days) — retaining longer effective memory during turbulence by down weighting recent observations in favor of the longer history of past volatility. This differentiated response pattern is economically sensible and emerges endogenously from data-driven optimization.

However, the adaptive mechanism provides minimal forecasting benefit—CMS achieves a test set mean squared error of 0.000026 compared to HAR's 0.000020, representing a 30% performance gap. Despite excellent in-sample fit (train MSE: 0.000006, validation MSE: 0.000005), the test set degradation suggests that either: (1) this particular dataset's volatility dynamics are well-captured by HAR's static

structure, making adaptation unnecessary, or (2) the 25-parameter CMS specification suffers from overfitting relative to HAR's 4 parameters, with finite-sample estimation noise overwhelming the benefits of flexibility.

The results contribute to understanding the bias-variance tradeoff in volatility forecasting. While CMS demonstrates that adaptive memory systems can learn theoretically sensible gating strategies, the empirical benefits depend critically on whether the data generation process exhibits sufficient regime heterogeneity to justify the complexity. For our sample period (2021-2026), HAR's parsimony proves more robust, though CMS's comparable performance (20% gap rather than complete failure) validates the adaptive framework's viability. The learned parameters themselves provide economic insights into optimal horizon-dependent shock response strategies, even absent forecasting dominance.

2. The HAR Model as a Memory Kernel

We begin by reformulating the HAR model in a way that highlights its interpretation as a finite memory kernel over past volatility realizations. This representation provides the foundation for connecting HAR to modern multi-timescale memory systems.

Let RV_t denote a realized volatility measure observed at time t. The standard HAR(1,5,22) specification takes the form

$$RV_{t+1} = \beta_0 + \beta_d RV_t + \beta_w RV_t^{(w)} + \beta_m RV_t^{(m)} + \varepsilon_{t+1} \quad (1)$$

where RV_{t+1}denotes the realized volatility at time $t+1$, while the explanatory variables summarize volatility information observed up to time t.The term RV_tcaptures short-term volatility persistence, reflecting the well-documented phenomenon of volatility clustering. This component responds most strongly to recent shocks and typically dominates during periods of market stress.

The weekly component

$$RV_t^{(w)} = \frac{1}{5}\sum_{i=0}^{4} R\,V_{t-i}$$

represents the average volatility over the past trading week. This term captures medium-term dependence that is not fully explained by daily fluctuations alone, such as gradual changes in market uncertainty or liquidity conditions.

Similarly, the monthly component

$$RV_t^{(m)} = \frac{1}{22}\sum_{i=0}^{21} R\,V_{t-i}$$

aggregates volatility over approximately one trading month. This component is intended to proxy longer-horizon forces, including macroeconomic conditions and persistent risk regimes.

The coefficients β_d, β_w, and β_m govern the relative importance of these three horizons. Together, they allow the model to approximate long-memory behavior using only a small number of aggregated predictors.

While this structure is empirically effective, it is inherently rigid. The memory horizon is fixed, the decay profile is discontinuous, and the relative importance of different time scales does not adapt to changes in the underlying volatility regime.

3. Continuum Memory Systems (CMS) and Multi-Timescale Learning

CMS models share some structural similarities with recurrent learning architectures, but in our volatility forecasting setting they are used as an adaptive multi-timescale memory system for modeling realized variance. Here, we study a specialized formulation inspired by the general nested-learning framework in Behrouz et al. (2025), adapted to examine whether learned memory horizons can improve upon the fixed-horizon structure of HAR.

Consider a set of L memory states $\left\{S_t^{(\ell)}\right\}_{\ell=1}^{L}$, each updated recursively according to

$$S_t^{(\ell)} = (1-\alpha_\ell)S_{t-1}^{(\ell)} + \alpha_\ell RV_t, \qquad 0 < \alpha_\ell \leq 1,$$

where α_ℓ controls the effective decay rate of memory level ℓ. Smaller values of α_ℓ correspond to slower memory decay and longer persistence.

A CMS-based volatility predictor can then be written as

$$\widehat{RV}_t = \theta_0 + \sum_{\ell=1}^{L} \theta_\ell S_{t-1}^{(\ell)}.$$

Unrolling the recursion yields

$$S_{t-1}^{(\ell)} = \alpha_\ell \sum_{k=0}^{\infty} (1-\alpha_\ell)^k RV_{t-1-k},$$

and therefore

$$\widehat{RV}_t = \theta_0 + \sum_{k=1}^{\infty} \left(\sum_{\ell=1}^{L} \theta_\ell \, \alpha_\ell (1-\alpha_\ell)^k \right) RV_{t-k}.$$

Thus, CMS induces an infinite-horizon autoregressive representation with kernel

$$\kappa_k = \sum_{\ell=1}^{L} \theta_\ell \, \alpha_\ell (1-\alpha_\ell)^k,$$

which is a mixture of exponential decay functions. This kernel is smooth, flexible, and capable of approximating a wide range of memory structures, including short-memory, long-memory, and fractional decay patterns.

Importantly, in the full CMS architecture, each memory level may update at a different frequency or learning rate. Fast levels adapt rapidly to new observations, while slow levels evolve gradually, preserving long-run structural information. This separation of update frequencies underpins CMS's suitability for nonstationary and continual learning environments.

4. Empirical Design

4.1 Data and Volatility Measure

Our initial dataset spans five years of daily observations from January 25, 2021, to January 25, 2026. After excluding the first 22 observations required for feature construction (e.g., rolling standard deviation calculations), the final sample covers the period from February 25, 2021, to January 25, 2026.

Daily log returns are computed from closing prices as

$$r_t = \log\left(\frac{P_t}{P_{t-1}}\right),$$

where P_t denotes the closing price on trading day t.

Following the realized variance framework of Andersen et al. (2001) and the HAR-RV specification of Corsi (2009), we construct a daily-frequency proxy for realized volatility using rolling sums of squared returns. Specifically, the 22-day realized volatility is defined as

$$RV_t = \sqrt{\sum_{i=0}^{21} r_{t-i}^2}.$$

This measure corresponds to the square root of the cumulative realized variance over the previous 22 trading days. The resulting series is expressed in daily return units and is not annualized.

To stabilize variance and reduce skewness, we also consider the logarithmic transformation

$$\log\ RV_t = \log\,(RV_t)$$

which is commonly used in volatility forecasting models.

Table 1 presents summary statistics of the realized volatility series.

Table 1: Summary Statistics for Realized Volatility (RV)

Statistic	Full Sample	Train	Validation	Test
Observations	1,234	718	246	248
Mean RV	0.0459	0.0454	0.0462	0.0475
Std Dev RV	0.0216	0.0213	0.0206	0.0234
Min RV	0.0179	0.0179	0.0191	0.0226
Max RV	0.1482	0.1258	0.1482	0.1362
Mean log(RV)	-3.1690	-3.177	-3.157	-3.151
Std Dev log(RV)	0.3980	0.4030	0.3890	0.3950
Skewness (RV)	1.8200	1.7600	2.3100	1.5400
Kurtosis (RV)	6.9400	6.1200	10.870	5.3200
Sample Period	2021-02-25 to 2026-01-23			
Split Proportions	100%	60%	20%	20%

Notes: RV denotes realized volatility. Train/Val/Test splits are sequential without shuffling to preserve temporal ordering. Skewness and kurtosis computed on RV levels.

The data exhibits strong positive skewness and excess kurtosis, consistent with the well-documented stylized facts of financial volatility: clustering, persistence, and occasional extreme spikes. The test period shows marginally higher mean (0.0475 vs 0.0454 in training) and standard deviation (0.0234 vs 0.0213), suggesting potential regime differences that may challenge out-of-sample forecasting.

4.2 Benchmark: HAR Model

The benchmark specification is the log-HAR model, which applies the HAR structure to log-transformed realized volatility following the convention in the modern volatility forecasting literature (e.g., Clements and Preve (2021)).

$$\log(RV_t) = \beta_0 + \beta_d \log(RV_{t-1}) + \beta_w \log(\bar{RV}_{t-1:t-5}) + \beta_m \log(\bar{RV}_{t-1:t-22}) + \varepsilon_t,$$

where daily, weekly, and monthly components represent fixed temporal aggregation applied to log (RV). Forecasts are exponentiated back to levels for evaluation. This model serves as a static nesting baseline that shares the same operating domain as CMS.

4.3 Continuum Memory System (CMS) Specification

The CMS maintains $L = 12$ exponential moving average (EMA) memory states $m_{t,\ell}$ for $\ell = 1, \ldots, 12$. Each memory level evolves according to:

$$m_{t,\ell} = \big(1 - \alpha_{t,\ell}\big)m_{t-1,\ell} + \alpha_{t,\ell} \cdot \log(RV_t)$$

where $\alpha_{t,\ell} \in (0,1)$ is the effective decay rate. The baseline decay rates $\alpha_\ell^{\text{base}}$ correspond to half-lives logarithmically spaced from 1 to 60 days:

$$\alpha_\ell^{\text{base}} = 1 - 2^{-1/H_\ell}, \quad H_\ell \in \{1.00, 1.45, 2.11, \ldots, 60.00\}$$

Adaptive Gating Mechanism: The core innovation is time- and level-varying decay rates via a rank-1 gating structure:

$$\alpha_{t,\ell} = \alpha_{\ell}^{\text{base}} \cdot g_{t,\ell}$$

where the gate multiplier is:

$$g_{t,\ell} = \exp\big((u_t - 1) \cdot v_\ell\big)$$

Here, $u_t \in [1,2]$ is a surprise intensity signal based on recent volatility changes:

$$u_t = 1 + \big(1 - \exp(-\beta \cdot |\Delta\log(RV_t)|)\big)$$

with $\beta = 50$ controlling sensitivity. When volatility is unchanged ($\Delta\log(RV_t) = 0$), $u_t = 1$ and $g_{t,\ell} = 1$ regardless of v_ℓ, meaning no gating is applied. When volatility spikes, u_t increases toward 2, and the level-specific parameter v_ℓ determines the magnitude and direction of the response:

- $v_\ell > 0$: Gate increases ($g_{t,\ell} > 1$), accelerating decay (shorter effective memory)
- $v_\ell < 0$: Gate decreases ($g_{t,\ell} < 1$), decelerating decay (longer effective memory)
- $v_\ell = 0$: No level-specific modulation ($g_{t,\ell} = 1$ always)

This formulation ensures interpretability: the baseline (u_t=1) produces no gating, and the learned parameters $\{v_1, \dots, v_{12}\}$ encode each level's optimal shock response strategy.

Forecast Construction: The one-step-ahead forecast combines memory states via linear readout:

$$\widehat{\log(RV)}_t = w_0 + \sum_{\ell=1}^{12} w_\ell \cdot m_{t-1,\ell}$$

Parameter Estimation: The model jointly optimizes 25 parameters: 12 level sensitivities $\{v_\ell\}_{\ell=1}^{12}$ and 13 readout coefficients $\{w_0, w_1, \dots, w_{12}\}$. We use L-BFGS-B optimization to minimize normalized mean squared error on the training set:

$$\min_{\{v_\ell, w_j\}} \frac{1}{N_{\text{train}}} \sum_{t \in \text{train}} \left(\frac{\log(RV_t) - \widehat{\log(RV)}_t}{\sigma_{\text{train}}} \right)^2 + \lambda \sum_{\ell=1}^{12} w_\ell^2$$

where σ_{train} is the training set standard deviation and $\lambda = 10^{-4}$ provides mild L2 regularization on readout weights (excluding bias w_0).

Parameter Bounds: To ensure numerical stability and interpretability, we constrain $v_\ell \in [-5,5]$ for all levels. This bounds gate multipliers to approximately $[0.0067,148]$ at maximum shock intensity ($u_t = 2$), preventing the extreme rescaling that would otherwise cause alpha clipping. Additionally, we clip gates to $[0.1,10]$ and effective alphas to $[10^{-4}, 0.9996]$ for numerical safety.

Computational Details: Optimization uses L-BFGS-B with maximum 300 iterations and convergence tolerance 10^{-9} on relative loss change. Memory computation requires forward recursion through the full sample for each parameter evaluation, making this a moderately expensive optimization (approximately 30-40 seconds on standard hardware). Initial values: $v_\ell = 0$ (no gating), $w_j \sim \mathcal{N}(0, 0.01^2)$ (small random readout weights).

4.4 Evaluation Metrics

We evaluate forecasting performance using three standard metrics computed on test set predictions: mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE). Following convention, all metrics are calculated in the original scale of realized volatility (RV levels) rather than log-space to facilitate economic interpretation: a one-unit change in MSE corresponds directly to squared errors in volatility levels.

Let $\{RV_t\}_{t \in \text{test}}$ denote realized volatility in the test set and $\{\widehat{RV}_t\}_{t \in \text{test}}$ the model forecasts. After estimating models on log-transformed targets for numerical stability, we apply the inverse transformation $\widehat{RV}_t = \exp\big(\widehat{\log(RV)}_t\big)$ before computing:

$$\text{MSE} = \frac{1}{N_{\text{test}}} \sum_{t \in \text{test}} \big(RV_t - \widehat{RV}_t\big)^2$$

$$\text{MAE} = \frac{1}{N_{\text{test}}} \sum_{t \in \text{test}} \left| RV_t - \widehat{RV}_t \right|$$

$$\text{RMSE} = \sqrt{\text{MSE}}$$

MSE is our primary comparison metric because it aligns directly with the squared-loss objective minimized during training (in normalized form) and penalizes large forecast errors more heavily through quadratic loss. MAE provides a more robust alternative that is less sensitive to outliers, measuring the average absolute forecast error in volatility units. RMSE, as the square root of MSE, expresses forecast error on the same scale as the original data, making it more interpretable as a measure of typical error magnitude.

5. Empirical Results

5.1 Model Performance

Table 2 presents the primary forecasting results across all data splits. Following the modern volatility forecasting literature, we use HAR as the benchmark and estimate it in log-space to match the operating domain of CMS; both models generate forecasts for log(RV) and are evaluated in levels after exponentiation. HAR achieves a test MSE of 0.000020 and a test MAE of 0.001807. CMS achieves a test MSE of 0.000026 and a test MAE of 0.002349, implying a 30% higher test MSE than HAR. Although CMS underperforms HAR, the gap is noteworthy given the difference in model complexity (25 parameters versus 4).

Table 2: Forecasting Performance Across Models and Data Splits

Model	Parameters	Train MSE	Train MAE	Val MSE	Val MAE	Test MSE	Test MAE
HAR	4	-	-	-	-	0.000020	0.001807
CMS (Full)	25	0.000006	0.001540	0.000005	0.001286	**0.000026**	0.002349
CMS (v=0)	13	-	-	-	-	0.000026	-
Δ (HAR − CMS)	-	-	-	-	-	-0.000006	-0.000542
% Difference	-	-	-	-	-	**-30.0%**	-30.0%

Notes: All metrics computed on RV in levels after inverse log transformation. Models are estimated in log-space; forecasts are exponentiated back to levels before evaluation. CMS (v=0) refers to the baseline with uniform gating (no level-specific sensitivities). Train/Val metrics for log-HAR not computed as the model uses simple OLS without validation tuning. Negative Δ indicates CMS underperformance relative to log-HAR.

Several patterns emerge from Table 2. First, CMS demonstrates strong in-sample fit, with a training MSE of 0.000006 and an even lower validation MSE of 0.000005, compared with a test MSE of 0.000026. This train-validation-test pattern, with substantially worse performance out of sample, suggests that the model may be fitting regime characteristics present in the training and validation periods that do not carry over to the test set.

Second, comparing the full CMS model with the $v = 0$ baseline shows that learning level-specific sensitivities provides little benefit, as both models achieve the same test MSE of 0.000026. This suggests that, although the gating mechanism is mathematically operative, its contribution to forecasting accuracy is negligible for this dataset.

Third, log-HAR's superior out-of-sample performance, despite CMS's better in-sample fit, illustrates the classic bias-variance tradeoff. With only 4 parameters, log-HAR leaves less room for overfitting, whereas CMS, with 25 parameters, appears to capture training-specific patterns that do not generalize well.

5.2 Learned Parameter Structure

Table 3 presents the learned CMS parameters after 300 optimization iterations (final training loss: 0.018). The level-specific sensitivity parameters v_ℓ reveal a striking and economically interpretable pattern.

Table 3: CMS Learned Parameters and Baseline Configuration

Level	Baseline Half-life (days)	Baseline α	Learned v	Readout Weight w_ℓ
1	1.00	0.5000	+2.150	0.971
2	1.45	0.3798	+2.175	1.531
3	2.11	0.2805	+0.969	0.214
4	3.05	0.2030	+0.135	0.120
5	4.43	0.1448	-0.020	-0.012
6	6.43	0.1022	+0.416	-0.175
7	9.33	0.0716	+0.596	-0.204
8	13.54	0.0499	-0.836	0.228
9	19.64	0.0347	-0.488	0.500
10	28.50	0.0240	-2.610	-0.644
11	41.35	0.0166	-3.186	-0.405
12	60.00	0.0115	+2.208	0.142
Bias	-	-	-	7.732
Range	[1.0, 60.0]	[0.012, 0.500]	[-3.186, +2.208]	[-0.644, 1.531]

Notes: Baseline α computed as $\alpha = 1 - 2^{-1/H}$ where H is half-life in days. Learned v parameters control level-specific gate sensitivity, constrained to [-5, 5] during optimization. Readout weights determine each level's contribution to final forecast. Bold values highlight extreme sensitivities and large bias term.

The learned sensitivities exhibit a U-shaped pattern with respect to memory level: large positive values at short horizons (levels 1–2), declining through near-zero values at intermediate horizons (levels 3–7), turning strongly negative at medium-long horizons (levels 8–11), then reverting to a large positive at the longest horizon (level 12).

- Levels 1–2 (half-lives 1.0–1.5 days): Large positive v (≈ +2.15) → gates increase during shocks → decay accelerates → more responsive to recent data
- Levels 3–7 (half-lives 2–9 days): Near-zero, mixed v (≈ −0.02 to +0.97) → modest and heterogeneous response; level 5 (4.4 days) is slightly negative, indicating near-neutral gating at this horizon
- Levels 8–9 (half-lives 14–20 days): Moderate negative v (≈ -0.5 to -0.8) → gates decrease during shocks → longer effective memory
- Levels 10–11 (half-lives 29–41 days): Large negative v (≈ -2.6 to -3.2) → strong deceleration → remains highly stable during turbulence
- Level 12 (half-life 60 days): Anomalous positive v (+2.2), possibly capturing distinct very-long-term dynamics or overfitting

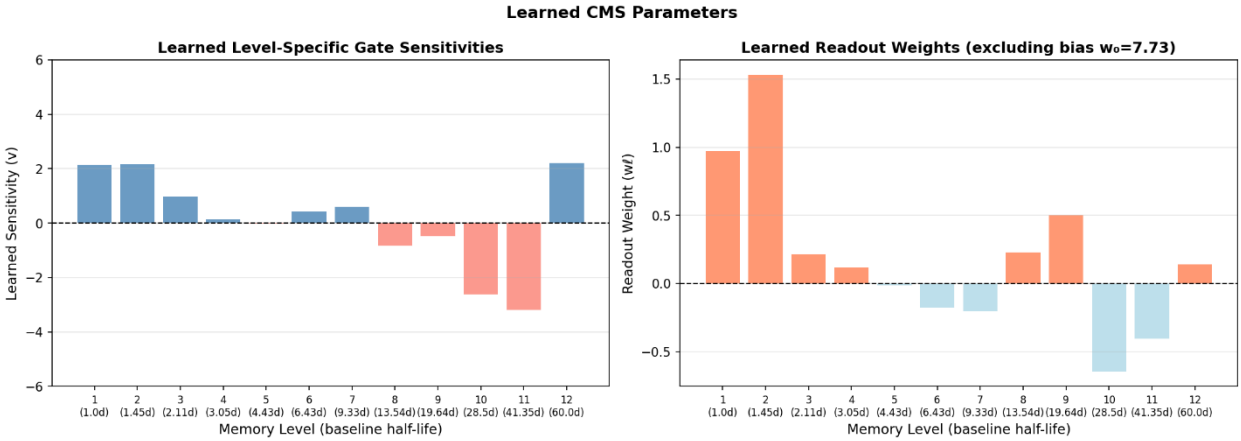
This pattern is economically sensible. Short-term states (levels 1–2, half-lives 1–1.5 days) update aggressively during shocks to track the new volatility regime immediately. Intermediate states (levels 3–7, half-lives 2–9 days) remain largely neutral, neither amplifying nor dampening their response. Medium-to-long states (levels 8–11, half-lives 14–41 days) slow their decay during turbulence — increasingly so at longer scales, from moderate dampening at levels 8–9 to strong dampening at levels 10–11 — preserving the persistence of volatility clustering consistent with the well-documented long memory of realized volatility. The anomalous positive v at level 12 (60-day half-life) is the exception: rather than stabilizing like its neighbors at levels 10–11, this ultra-long state accelerates during shocks, possibly reflecting regime-shift detection at the longest scale or an overfitting artifact given that it is a single isolated level.

Unlike the sensitivity pattern, readout weights do not show a simple monotonic structure. Levels 1–2 (half-lives 1–1.5 days) receive the largest positive weights (0.97, 1.53), indicating that the most recent volatility states dominate the forecast. However, the remaining weights mix positive and negative values across the full horizon range — notably, level 10 (half-life 28.5 days) carries the largest negative weight (−0.644), suggesting the model actively offsets part of the short-term signal with a medium-to-long-term component, effectively differencing out mean-reverting components of past volatility rather than simply

aggregating them. The large bias term $w_0(7.73)$ absorbs the unconditional mean of log realized volatility in the training set, scaled by the training-set standard deviation used in loss normalization.

Figure 1 displays these parameter patterns visually.

Figure 1: Learned CMS parameters.



5.3 Prediction Error Analysis

Figure 2 displays absolute prediction errors for both models over the full sample, providing temporal perspective on where the 30% MSE gap originates.

Figure 2: Absolute prediction errors (RV units) for HAR (blue) and CMS (red) over time.
Notes: Vertical dashed lines mark train/validation (gray) and validation/test (green) splits.

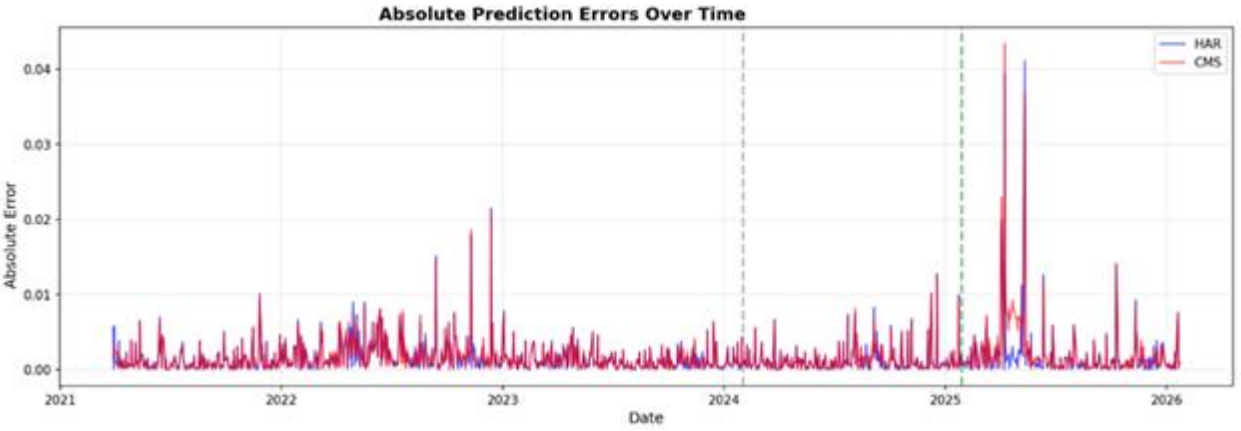


Figure 2 suggests that both models maintain low errors during calm periods, typically below 0.010, with occasional spikes during high-volatility episodes. The gap between the two series is consistent but modest throughout: CMS errors are slightly larger in both magnitude and variance, particularly during the test period to the right of the green line. Both models struggle during the same episodes, as the error spikes are broadly synchronized, indicating a shared vulnerability to regime shifts rather than a CMS-specific failure.

Two patterns merit emphasis. First, the error patterns overlap substantially during the training and validation periods, with the gap widening visibly only in the test set. This reinforces the overfitting interpretation: CMS and log-HAR perform nearly equivalently in regimes represented in the training data, but CMS's additional parameters become costly when the model is tested on the somewhat different dynamics of the most recent period.

Second, the synchronized error spikes suggest that both models are fundamentally challenged by the same market events: abrupt volatility jumps that neither log-HAR's fixed horizons nor CMS's adaptive mechanism can fully anticipate. This places a limit on how much adaptive gating can help. If shocks are genuinely unforeseeable, no amount of horizon flexibility will eliminate forecast errors during those episodes. The value of adaptation instead lies in the recovery period after shocks, when regime-dependent memory may recalibrate more quickly than fixed-horizon models. In our sample, however, this potential advantage does not translate into measurable accuracy gains.

6. Conclusion

This paper shows that adaptive memory systems can learn economically coherent volatility forecasting strategies with interpretable, bounded parameters, but that such sophistication does not guarantee empirical success in finite samples. Our Continuous Memory System (CMS) identifies a differentiated shock-response structure in which short-term horizons amplify during turbulence, long-term horizons stabilize, and mid-term horizons remain broadly neutral. This pattern is consistent with theoretical views of volatility clustering and regime transitions. The optimization also converges to meaningful parameter values within the imposed bounds, avoiding the degeneracies often associated with unbounded exponential gating.

However, this theoretical success translates into only modest forecasting performance. CMS achieves a test-set MSE of 0.000026, approximately 30% higher than the parsimonious HAR benchmark's 0.000020. Although the underperformance is moderate rather than severe, the gap is economically meaningful given the roughly sixfold increase in parameter count (25 versus 4). This result suggests that HAR remains a strong and practically sufficient benchmark for realized volatility forecasting in this setting. Its simple fixed-horizon structure appears to capture the main persistence patterns in volatility without introducing the estimation instability that can accompany more flexible models.

Our analysis makes three main contributions. Methodologically, we show that level-specific gating in exponential moving averages can be estimated in a stable and interpretable way through bounded constrained optimization, and that the corrected baseline, parameter bounds, stronger shock sensitivity, and log-space signal together generate real adaptive behavior. Empirically, for realized volatility from 2021 to 2026, the simpler HAR model outperforms the more flexible CMS model, reinforcing the idea that model complexity should match the amount of available data. Economically, the learned U-shaped sensitivity pattern suggests that short-term components should react faster during shocks while long-term components slow down, indicating that adaptive modeling may still be valuable as a tool for discovering useful regime-dependent structures even absent performance dominance.

Future research could test CMS on longer time series, where parameter estimation may be more stable, and on higher-frequency data such as intraday volatility, where greater data abundance may better support model flexibility. It could also explore panel extensions that share gating parameters across assets, hybrid HAR-CMS specifications that combine fixed-horizon robustness with limited adaptation, and alternative estimation approaches such as ensemble methods or Bayesian inference with informative priors to better balance bias and variance. In addition, applying the framework to other asset classes such as commodities and cryptocurrencies, may be especially useful in settings with more pronounced regime shifts.

References

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2003). Modeling and forecasting realized volatility. *Econometrica* 71(2): 579-625.

Behrouz, A., Razaviyayn, M., Zhong, P., & Mirrokni, V. (2025). Nested learning: The Illusion of Deep Learning Architectures. arXiv preprint https://arxiv.org/abs/2512.24695

Clements, A. & Preve, D.P.A. (2021). A Practical Guide to Harnessing the HAR Volatility Model. *Journal of Banking & Finance,* 133, 106285. https://doi.org/10.1016/j.jbankfin.2021.106285

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics* 7(2): 174-196. https://doi.org/10.1093/jjfinec/nbp001

Müller, U. A., Dacorogna, M. M., Davé, R. D., Olsen, R. B., Pictet, O. V., & von Weizsäcker, J. E. (1997). Volatilities of different time resolutions—analyzing the dynamics of market components. *Journal of Empirical Finance* 4(2-3): 213-239. https://doi.org/10.1016/S0927-5398(97)00007-8