



Research article

Uncertainty calibrated volatility forecasting through regime detection and adaptive ensemble modeling

Khalid Jeaab^{1,*}, Smaaine Ouaharahe² and Youness Saoudi³

¹ Sultan Moulay Slimane University, Economics and Management Laboratory, Khouribga, Morocco

² Ibn Tofail University, Organization Economics and Management Laboratory, Kenitra, Morocco

³ Ibn Tofail University, Advanced Systems Engineering Laboratory, Kenitra, Morocco

* **Correspondence:** Email: jeaab.khalid.feg@usms.ac.ma.

A. Technical implementation details

A.1. Feature engineering specifications

Table A1 provides complete specifications for all 42 features used in RADE-UQ, including the economic rationale for each feature category as motivated by the established realized volatility literature.

Table A1. Complete feature specifications with economic rationale

Category	Feature	Description	Economic rationale
HAR	$RV_t^{(d)}$	Daily realized volatility	Captures daily volatility persistence; foundation of HAR-RV model (Corsi, 2009). Dominant predictor at short horizons.

Continued on next page

Table A1 (*continued*)

Category	Feature	Description	Economic rationale
	$RV_t^{(w)}$	Weekly average (5 days): $\frac{1}{5} \sum_{i=0}^4 RV_{t-i}$	Weekly volatility component; approximates short-memory mean reversion. Captures trader heterogeneity at weekly frequency (Corsi, 2009).
	$RV_t^{(m)}$	Monthly average (22 days): $\frac{1}{22} \sum_{i=0}^{21} RV_{t-i}$	Long-memory approximation via simple AR structure; captures institutional rebalancing cycles and macro volatility regimes (Corsi, 2009).
Momentum	$r_{t,5}$	5-day cumulative return	Short-term momentum; negative correlation with volatility via leverage effect (Black, 1976; Christie, 1982).
	$r_{t,10}$	10-day cumulative return	Medium-term return trend; captures two-week momentum cycles documented in equity markets (Jegadeesh and Titman, 1993).
	$r_{t,20}$	20-day cumulative return	Monthly return momentum; proxy for risk-appetite regime changes affecting market-wide volatility (Ang et al., 2006).
	RSI_t	Relative strength index (14-day)	Overbought/oversold signal; elevated RSI associated with complacency and subsequent volatility increases (Brown and Jennings, 1989).
	$MACD_t$	Moving average convergence divergence	Trend-reversal indicator; crossovers signal regime transitions associated with volatility clustering (Brock et al., 1992).
	BB_t	Bollinger band position: $\frac{P_t - MA_{20}}{2\sigma_{20}}$	Price position relative to volatility bands; extreme values predict mean reversion and volatility spikes (Bollinger, 1992).

Continued on next page

Table A1 (*continued*)

Category	Feature	Description	Economic rationale
Volatility clustering	$\sigma_{t,5}$	5-day rolling std. deviation	Short-term volatility clustering (Engle, 1982; Bollerslev, 1986); primary ARCH effect at weekly horizon.
	$\sigma_{t,10}$	10-day rolling std. deviation	Bi-weekly volatility persistence; captures medium-term GARCH dynamics beyond the HAR-RV lags.
	$\sigma_{t,20}$	20-day rolling std. deviation	Monthly volatility baseline: benchmark for normalizing other features and detecting anomalous volatility states.
	ATR_t	Average true range (14-day)	Range-based volatility estimator; more robust to gaps and limit moves than close-to-close estimators (Parkinson, 1980).
	GK_t	Garman-klass volatility estimator	Efficient OHLC-based volatility estimate; up to 7.4× more efficient than close-to-close variance (Garman and Klass, 1980).
Volume	$V_t/\bar{V}_{t,20}$	Relative volume vs. 20-day moving average	Volume surprise; abnormal volume precedes volatility increases and regime transitions (Karpoff, 1987; Lamoureux and Lastrapes, 1990).
	ΔV_t	Volume change: $\frac{V_t - V_{t-1}}{V_{t-1}}$	Day-over-day volume acceleration; sudden increases signal arrival of informed traders and imminent volatility (Blume et al., 1994).
	OBV_t	On-balance volume (cumulative)	Cumulative volume flow; divergence between price and OBV predicts trend reversals and associated volatility bursts (Granville, 1963).

Continued on next page

Table A1 (*continued*)

Category	Feature	Description	Economic rationale
	$VWAP_t$	Volume-weighted average price	Institutional benchmark price; deviations from VWAP indicate order imbalance and intraday volatility risk (Berkowitz et al., 1988).
Jump components	$\mathbb{I}\{ r_t > 3\sigma_t\}$	Jump indicator (3σ threshold)	Detects discontinuous price moves; jump arrival is a leading indicator of elevated volatility persistence (Barndorff-Nielsen and Shephard, 2006; Andersen et al., 2007).
	BV_t	Bipower variation: $\sum_{i=1}^{M-1} r_{t,i} r_{t,i+1} $	Jump-robust volatility estimator; converges to integrated variance in the presence of jumps (Barndorff-Nielsen and Shephard, 2004).
	JR_t	Jump ratio: $\frac{RV_t - BV_t}{RV_t}$	Proportion of total variance due to jumps; high JR predicts subsequent volatility mean reversion as jump uncertainty resolves (Andersen et al., 2007).
Leverage Effects	$r_t^2 \cdot \text{sign}(r_t)$	Signed squared return	Direct leverage effect measure; negative returns amplify subsequent volatility (Black, 1976; Nelson, 1991; EGARCH).
	$r_t^- \cdot RV_t$	Negative return $\times$ volatility interaction	Non-linear leverage interaction; captures asymmetric volatility amplification specifically during downside moves (Glosten et al., 1993; GJR-GARCH).
	Asym_t	Asymmetric volatility: $\frac{\sum_{i:r_i < 0} r_i^2}{\sum_i r_i^2}$	Rolling downside variance ratio; values > 0.5 indicate negative return dominance and elevated tail risk (Ang et al., 2006).
	$\text{Skew}_{t,20}$	20-day rolling skewness	Return distribution asymmetry; negative skewness predicts crash risk and volatility spikes (Harvey and Siddique, 2000).

Continued on next page

Table A1 (*continued*)

Category	Feature	Description	Economic rationale
	$Kurt_{t,20}$	20-day rolling kurtosis	Tail heaviness; excess kurtosis signals fat-tailed return distributions with elevated jump risk (Christoffersen, 2012).
Macroeconomic (FRED)	VIX_t	CBOE volatility index (FRED: VIXCLS)	Forward-looking implied volatility; fear gauge reflecting market expectations and risk aversion (Whaley, 2000). Strongest macro predictor of realized volatility.
	$UNRATE_t$	Unemployment rate (FRED: UNRATE)	Business cycle indicator; rising unemployment signals economic contraction and elevated macro uncertainty (Hamilton, 1989).
	DFF_t	Federal funds effective rate (FRED: DFF)	Monetary policy stance; rapid rate changes generate bond and equity volatility spikes (Bernanke and Kuttner, 2005).
	$T10Y2Y_t$	Term spread: 10Y minus 2Y Treasury (FRED: T10Y2Y)	Yield curve slope; inversion predicts recession and associated equity volatility regime shifts (Estrella and Mishkin, 1998).
	ΔCPI_t	CPI growth year-over-year (FRED: CPIAUCSL)	Inflation regime indicator; high or accelerating inflation generates monetary policy uncertainty and volatility (Schwert, 1989).
	$HYSpread_t$	High-yield credit spread (FRED: BAMLH0A0HYM2)	Credit risk and financial stress measure widen ahead of equity volatility episodes and liquidity crises (Campbell and Taksler, 2003).
Regime detection	$\sigma(RV_t)_{22}$	Volatility of volatility (22-day rolling)	Second-order uncertainty measure; high VoV signals impending regime transition (Bollerslev et al., 2009; VRP literature).

Continued on next page

Table A1 (*continued*)

Category	Feature	Description	Economic rationale
	$\rho_1(RV_t)$	Autocorrelation at lag 1	Short-term volatility persistence; declining lag-1 autocorrelation signals regime instability (Hamilton, 1989).
	$\rho_5(RV_t)$	Autocorrelation at lag 5	Weekly autocorrelation structure; captures medium-term regime persistence beyond daily mean reversion.
	$\rho_{22}(RV_t)$	Autocorrelation at lag 22	Monthly autocorrelation: proxies for long-memory behavior and slow-moving macro volatility cycles (Ding et al., 1993).
	Percentile _{<i>t</i>}	Volatility percentile rank (252-day trailing window)	Relative volatility state within its historical distribution; directly maps to low/medium/high regime labels without parametric assumptions.
	$\mathbf{1}\{\text{High} \rightarrow \text{Low}\}$	High-to-low regime transition indicator	Flags immediate post-stress mean reversion; volatility typically overshoots during transitions, requiring adaptive interval width (Schwert, 1990).
	$\mathbf{1}\{\text{Low} \rightarrow \text{High}\}$	Low-to-high regime transition indicator	Early warning of volatility escalation precedes VaR breaches and requires preemptive interval widening for risk management.
	Duration _{<i>t</i>}	Days in current regime	Regime age; longer-duration regimes are more likely to persist (regime inertia) but also accumulate transition risk (Hamilton, 1989).
	Hurst _{<i>t</i>}	Hurst exponent (60-day rolling window)	Long-memory and mean-reversion indicator: $H > 0.5$ (trending), $H < 0.5$ (mean-reverting). Distinguishes persistent high-volatility regimes from transient spikes (Mandelbrot and Van Ness, 1968).

Total: 42 features (3 HAR + 23 market + 6 macro + 9 regime + 1 target)

A.1. Regime detection network architecture

A.1.1. Complete mathematical formulation

Input layer. Time series sequences $\mathbf{X}_t = [\mathbf{x}_{t-L+1}, \dots, \mathbf{x}_t] \in \mathbb{R}^{L \times d}$ where $L = 60$ is lookback window and $d = 42$ is feature dimension.

Bi-LSTM layer. Bidirectional LSTM processes sequences forward and backward. For each time step i , the LSTM cell computes

$$\mathbf{f}_i = \sigma(\mathbf{W}_f \mathbf{x}_i + \mathbf{U}_f \mathbf{h}_{i-1} + \mathbf{b}_f) \quad (\text{forget gate}) \quad (1)$$

$$\mathbf{i}_i = \sigma(\mathbf{W}_i \mathbf{x}_i + \mathbf{U}_i \mathbf{h}_{i-1} + \mathbf{b}_i) \quad (\text{input gate}) \quad (2)$$

$$\mathbf{o}_i = \sigma(\mathbf{W}_o \mathbf{x}_i + \mathbf{U}_o \mathbf{h}_{i-1} + \mathbf{b}_o) \quad (\text{output gate}) \quad (3)$$

$$\tilde{\mathbf{c}}_i = \tanh(\mathbf{W}_c \mathbf{x}_i + \mathbf{U}_c \mathbf{h}_{i-1} + \mathbf{b}_c) \quad (\text{cell candidate}) \quad (4)$$

$$\mathbf{c}_i = \mathbf{f}_i \odot \mathbf{c}_{i-1} + \mathbf{i}_i \odot \tilde{\mathbf{c}}_i \quad (\text{cell state}) \quad (5)$$

$$\mathbf{h}_i = \mathbf{o}_i \odot \tanh(\mathbf{c}_i) \quad (\text{hidden state}) \quad (6)$$

where $\sigma(\cdot)$ sigmoid activation, $\tanh(\cdot)$ is hyperbolic tangent is, and $\odot$ denotes element-wise multiplication.

Forward and backward passes

$$\vec{\mathbf{h}}_i = \text{LSTM}_{\text{forward}}(\mathbf{x}_i, \vec{\mathbf{h}}_{i-1}) \quad (7)$$

$$\overleftarrow{\mathbf{h}}_i = \text{LSTM}_{\text{backward}}(\mathbf{x}_i, \overleftarrow{\mathbf{h}}_{i+1}) \quad (8)$$

$$\mathbf{h}_i = [\vec{\mathbf{h}}_i; \overleftarrow{\mathbf{h}}_i] \in \mathbb{R}^{2h} \quad (9)$$

Temporal attention. Multi-head attention with $H = 4$ heads identifies important time steps

$$e_{ij}^{(h)} = \frac{(\mathbf{W}_Q^{(h)} \mathbf{h}_i)^\top (\mathbf{W}_K^{(h)} \mathbf{h}_j)}{\sqrt{d_k}} \quad (10)$$

$$\alpha_{ij}^{(h)} = \frac{\exp(e_{ij}^{(h)})}{\sum_{k=1}^L \exp(e_{ik}^{(h)})} \quad (11)$$

$$\mathbf{c}_i^{(h)} = \sum_{j=1}^L \alpha_{ij}^{(h)} (\mathbf{W}_V^{(h)} \mathbf{h}_j) \quad (12)$$

$$\mathbf{c}_i = \text{Concat}(\mathbf{c}_i^{(1)}, \dots, \mathbf{c}_i^{(H)}) \mathbf{W}_O \quad (13)$$

Feature attention. Attention over feature dimensions

$$s_k = \mathbf{W}_f \mathbf{c}_k + \mathbf{b}_f \quad (14)$$

$$\beta_k = \frac{\exp(s_k)}{\sum_{k'} \exp(s_{k'})} \quad (15)$$

$$\mathbf{z} = \sum_k \beta_k \mathbf{c}_k \quad (16)$$

Output Layer. Regime probabilities via softmax

$$\mathbf{p}_t = \text{softmax}(\mathbf{W}_o \mathbf{z} + \mathbf{b}_o) \in \mathbb{R}^K \quad (17)$$

where $K = 3$ "number of regimes" denotes low, medium, and high volatility.

Training Loss

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K y_i^{(k)} \log p_i^{(k)} + \lambda \|\boldsymbol{\theta}\|_2^2 \quad (18)$$

where $\mathbf{y}_i$ one-hot regime label is, $\lambda = 0.001$ controls L2 regularization and $\boldsymbol{\theta}$ denotes all trainable parameters.

A.1.2. Hyperparameters and training results

Table A2. RDN hyperparameters and training results

Category	Parameter	Value
Architecture	Input dimension	42
	Lookback window	60
	Hidden dimension	128
	Number of LSTM layers	2
	Number of attention heads	4
	Output dimension (regimes)	3
Training	Dropout rate	0.3
	Learning rate	0.001
	Optimizer	Adam ($\beta_1 = 0.9, \beta_2 = 0.999$)
	Batch size	32
	Early stopping patience	10 epochs
Results	Training accuracy	42.3%
	Validation accuracy	34.7%
	Test accuracy	33.1%
	Training time	4.8 hours

Note on classification performance: The test accuracy of 33.1% is effectively at the random baseline of 33.3% for a balanced three-class problem. The gap between training (42.3%) and test (33.1%) accuracy indicates overfitting to training-set regime patterns (see table A2), consistent with the label noise hypothesis described in Section 6.4. The ablation study (Table 4) confirms that this accuracy level does not produce RMSE gains relative to an equal-weight ensemble. See Section 6.5 for proposed remediation strategies.

A.2. Hyperparameter tuning

We employ Bayesian optimization via Tree-structured Parzen Estimator (TPE) for hyperparameter tuning, minimizing validation RMSE over 100 trials (see table A3).

Table A3. Hyperparameter search spaces and optimal values

Model	Parameter	Range	Optimal
Random forest	n_estimators	[50, 200]	100
	max_depth	[5, 20]	10
	min_samples_split	[2, 20]	5
LightGBM	n_estimators	[50, 300]	150
	learning_rate	[0.01, 0.2]	0.05
	max_depth	[3, 10]	6
	num_leaves	[15, 63]	31
	min_data_in_leaf	[10, 50]	20
Ensemble	α (regime weight)	[0.0, 1.0]	0.5
	β (uncertainty weight)	[0.0, 1.0]	0.3
	γ (performance weight)	[0.0, 1.0]	0.2

Note: Constraint $\alpha + \beta + \gamma = 1$ enforced during optimization.

A.3. Computational complexity analysis

Table A4 analyzes computational complexity and wall-clock time for each component.

Table A4. Computational complexity analysis

Component	Time complexity	Training	Inference
HAR-RV (50 bootstraps)	$O(n \cdot B \cdot d^2)$	0.8h	0.02s
Random Forest (50 bootstraps)	$O(n \log n \cdot T \cdot d \cdot B)$	31.2h	0.15s
LightGBM (50 bootstraps)	$O(n \cdot T \cdot d \cdot B)$	10.5h	0.08s
RDN	$O(n \cdot L \cdot h^2)$	4.8h	0.12s
Ensemble aggregation	$O(n \cdot M \cdot B)$	–	0.05s
Total	–	48.2h	0.42s

n : sample size (2,515), B : bootstrap iterations (50), d : features (42),

T : trees/estimators (100/150), L : lookback (60), h : hidden dim (128), M : models (3)

Scalability analysis. Training time scales linearly with dataset size and bootstrap iterations. For production deployment with daily updates, incremental learning could reduce training time by 80-90% through online weight updates without full model retraining.

B. Mathematical derivations

B.1. Uncertainty decomposition proof

For an ensemble of M models with predictions $\{\hat{y}_m\}_{m=1}^M$, weights $\{w_m\}_{m=1}^M$, and within-model variances $\{\sigma_m^2\}_{m=1}^M$, the total prediction variance decomposes as

$$\text{Var}[\hat{y}_{\text{ensemble}}] = \underbrace{\sum_m w_m (\hat{y}_m - \bar{y})^2}_{\text{Epistemic}} + \underbrace{\sum_m w_m \sigma_m^2}_{\text{Aleatoric}} \quad (19)$$

Proof. Let $\hat{y}_{\text{ensemble}} = \sum_m w_m \hat{y}_m$ be the ensemble prediction. The total variance

$$\text{Var}[\hat{y}_{\text{ensemble}}] = \mathbb{E}[(\hat{y}_{\text{ensemble}} - \mathbb{E}[\hat{y}_{\text{ensemble}}])^2] \quad (20)$$

$$= \mathbb{E} \left[\left(\sum_m w_m \hat{y}_m - \sum_m w_m \mathbb{E}[\hat{y}_m] \right)^2 \right] \quad (21)$$

$$= \mathbb{E} \left[\left(\sum_m w_m (\hat{y}_m - \mathbb{E}[\hat{y}_m]) \right)^2 \right] \quad (22)$$

Expanding the square

$$= \sum_m \sum_{m'} w_m w_{m'} \mathbb{E}[(\hat{y}_m - \mathbb{E}[\hat{y}_m])(\hat{y}_{m'} - \mathbb{E}[\hat{y}_{m'}])] \quad (23)$$

Under the assumption that model predictions are independent (ensured by bootstrap resampling with different random seeds)

$$= \sum_m w_m^2 \text{Var}[\hat{y}_m] \quad (24)$$

Decompose the variance of each model's prediction using bias-variance decomposition

$$\text{Var}[\hat{y}_m] = \mathbb{E}[(\hat{y}_m - \mathbb{E}[y])^2] \quad (25)$$

$$= \mathbb{E}[(\hat{y}_m - \mathbb{E}[\hat{y}_m] + \mathbb{E}[\hat{y}_m] - \mathbb{E}[y])^2] \quad (26)$$

$$= \underbrace{\mathbb{E}[(\hat{y}_m - \mathbb{E}[\hat{y}_m])^2]}_{\sigma_m^2 \text{ (aleatoric)}} + \underbrace{(\mathbb{E}[\hat{y}_m] - \bar{y})^2}_{\text{epistemic}} \quad (27)$$

Substituting back and using $\sum_m w_m = 1$

$$\text{Var}[\hat{y}_{\text{ensemble}}] = \sum_m w_m [\sigma_m^2 + (\hat{y}_m - \bar{y})^2] \quad (28)$$

$$= \underbrace{\sum_m w_m (\hat{y}_m - \bar{y})^2}_{\sigma_{\text{epistemic}}^2} + \underbrace{\sum_m w_m \sigma_m^2}_{\sigma_{\text{aleatoric}}^2} \quad (29)$$

□

B.2. Calibration multiplier optimization

Problem statement. Given uncalibrated prediction intervals $[\hat{y}_i - \hat{\sigma}_i, \hat{y}_i + \hat{\sigma}_i]$ with empirical coverage $\text{ECR}_0 \neq (1 - \alpha)$, find a multiplier c^* such that scaled intervals $[\hat{y}_i - c\hat{\sigma}_i, \hat{y}_i + c\hat{\sigma}_i]$ achieve target coverage $(1 - \alpha)$.

Objective function

$$c^* = \arg \min_c |\text{ECR}(c) - (1 - \alpha)| \quad (30)$$

where

$$\text{ECR}(c) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}\{y_i \in [\hat{y}_i - c\hat{\sigma}_i, \hat{y}_i + c\hat{\sigma}_i]\} \quad (31)$$

Algorithm 1 Calibration multiplier optimization

```

1: Input: Validation predictions  $\{\hat{y}_i\}$ , uncertainties  $\{\hat{\sigma}_i\}$ , true values  $\{y_i\}$ , target coverage  $(1 - \alpha)$ 
2: Output: Optimal multiplier  $c^*$ 
3: Initialize:  $c_{\min} \leftarrow 0.1$ ,  $c_{\max} \leftarrow 10$ ,  $\epsilon \leftarrow 0.001$ 
4: while  $|c_{\max} - c_{\min}| > \epsilon$  do
5:    $c_{\text{mid}} \leftarrow (c_{\min} + c_{\max})/2$ 
6:   Compute  $\text{ECR}(c_{\text{mid}}) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}\{y_i \in [\hat{y}_i - c_{\text{mid}}\hat{\sigma}_i, \hat{y}_i + c_{\text{mid}}\hat{\sigma}_i]\}$ 
7:   if  $\text{ECR}(c_{\text{mid}}) < (1 - \alpha)$  then
8:      $c_{\min} \leftarrow c_{\text{mid}}$  ▷ Widen intervals
9:   else
10:     $c_{\max} \leftarrow c_{\text{mid}}$  ▷ Narrow intervals
11:   end if
12: end while
13: return  $c^* = c_{\text{mid}}$ 

```

Optimization algorithm (binary search)

Convergence guarantee. Binary search guarantees convergence in $O(\log((c_{\max} - c_{\min})/\epsilon))$ iterations. For our parameter range $[0.1, 10]$ and tolerance $\epsilon = 0.001$, convergence occurs in approximately 13 iterations.

RADE-UQ results

- Pre-calibration: $c = 1.96$ (theoretical gaussian), ECR = 68.52%
- Post-calibration: $c^* = 4.398$, ECR = 94.97%
- Kupiec test: $p = 0.923$ (statistically valid coverage)

C. Additional results and robustness checks*C.1. Performance by volatility regime (detailed)*

Table C5 extends the main text analysis with additional metrics.

Table C5. Detailed performance by volatility regime

Regime	N	RMSE	MAE	R^2	QLIKE	Cov.	IWQ
Low	126	0.0118	0.0076	0.935	0.0065	55.6%	0.342
Medium	126	0.0127	0.0080	0.928	0.0078	73.0%	0.581
High	126	0.0163	0.0102	0.903	0.0124	77.0%	1.243
Mean	–	0.0136	0.0086	0.922	0.0089	68.5%	0.722
Std	–	0.0020	0.0011	0.014	0.0026	9.3%	0.377
CV	–	0.144	0.128	0.015	0.292	–	0.522
Regime stability index (RSI): 0.856							

Key observations

- RMSE increases 38% from low to high volatility, substantially less than volatility itself (150-200% increase)

- R^2 remains above 0.90 across all regimes, indicating consistent explanatory power
- Coverage improves in higher volatility regimes (55.6% $\rightarrow$ 77.0%), suggesting conservative interval scaling
- IWQ increases dramatically in high volatility (0.342 $\rightarrow$ 1.243), confirming exceptional adaptive capacity

C.2. Robustness to bootstrap sample size

We investigate sensitivity to bootstrap iterations $B \in \{10, 25, 50, 100\}$.

Table C6. Robustness to bootstrap sample size

B	RMSE	R^2	Coverage	Time (h)	Δ RMSE
10	0.0145	0.916	91.8%	10.2	+5.1%
25	0.0141	0.920	93.4%	24.8	+2.2%
50	0.0138	0.922	94.9%	48.2	–
100	0.0137	0.923	95.1%	95.6	–0.7%

Findings: Performance plateaus at $B = 50$; increasing to $B = 100$ yields marginal 0.7% RMSE improvement at 2 $\times$ computational cost. Our choice $B = 50$ balances accuracy and efficiency.

C.3. Alternative weighting schemes

Table C7 compares additional ensemble weighting strategies.

Table C7. Alternative ensemble weighting strategies

Strategy	HAR-RV	RF	LGB	Test RMSE
Equal (current)	33.0%	33.0%	34.0%	0.0138
RMSE-optimal	14.4%	39.9%	45.7%	0.0134
QLIKE-optimal	7.1%	37.5%	55.4%	0.0136
Inverse-variance	8.3%	42.1%	49.6%	0.0135
Sharpe-optimal	18.2%	35.6%	46.2%	0.0137
Stacking (ridge)	12.5%	38.8%	48.7%	0.0133
Conformal pred. (LightGBM)	–	–	100%	0.0098 [†]

[†] Conformal prediction achieves coverage 95.1% (IWQ = 0.740 vs. RADE-UQ 0.962); reported as UQ benchmark, not a weighting strategy.

Insights: Stacking with ridge regression meta-learners achieves the best performance (RMSE: 0.0133) but requires additional validation complexity. RMSE-optimal weights offer 2.4% improvement with simple implementation.

C.4. Subperiod analysis

We analyze performance across four subperiods to assess temporal stability.

Table C8. Performance across temporal subperiods

Period	Dates	N	RMSE	R^2
Q1	2022-09 to 2023-03	95	0.0132	0.927
Q2	2023-04 to 2023-09	95	0.0128	0.931
Q3	2023-10 to 2024-03	94	0.0141	0.918
Q4	2024-04 to 2024-12	94	0.0149	0.912
Full Test	2022-09 to 2024-12	378	0.0138	0.922

Temporal stability: Performance remains consistent across subperiods (RMSE CV: 6.3%), demonstrating robustness to non-stationarity. Slight degradation in Q4 coincides with elevated market volatility.

C.5. Expanding-window out-of-sample robustness

To complement the static train/validation/test split, we conduct an expanding-window evaluation to verify that RADE-UQ's test-set performance does not depend on the choice of evaluation window. Starting from the initial training set (70% of observations), we reestimate the full RADE-UQ framework every 63 trading days (approximately one quarter), adding new observations to the training set while maintaining a fixed 63-day rolling evaluation window. Bootstrap iterations are reduced to $B = 25$ per model for computational tractability.

Table C9. Expanding-window out-of-sample performance

Window	Dates	n	RMSE	R^2
W1	2022-09 to 2022-12	63	0.0134	0.929
W2	2023-01 to 2023-03	63	0.0136	0.926
W3	2023-04 to 2023-06	63	0.0131	0.931
W4	2023-07 to 2023-09	63	0.0138	0.922
W5	2023-10 to 2023-12	63	0.0142	0.917
W6	2024-01 to 2024-03	63	0.0139	0.921
Range		–	0.0131–0.0151	0.912–0.931
CV (%)		–	6.3	0.8

RMSE ranges from 0.0131 to 0.0151 across all windows (CV = 6.3%), and R^2 remains above 0.91 in every window. The slight performance degradation in later windows (W5–W6) coincides with elevated market volatility in the 2023–2024 period, consistent with the subperiod analysis in Table 13. These results confirm that the reported test-set performance is stable across evaluation windows and is not an artifact of a particularly favorable holding period.

D. Complete evaluation metric definitions

D.1. Point forecast accuracy metrics

Root mean squared error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \hat{y}_t)^2} \quad (32)$$

Mean absolute error (MAE):

$$\text{MAE} = \frac{1}{N} \sum_{t=1}^N |y_t - \hat{y}_t| \quad (33)$$

Coefficient of determination (R^2):

$$R^2 = 1 - \frac{\sum_t (y_t - \hat{y}_t)^2}{\sum_t (y_t - \bar{y})^2} \quad (34)$$

QLIKE loss (?):

$$\text{QLIKE} = \frac{1}{N} \sum_{t=1}^N \left[\frac{y_t}{\hat{y}_t} - \log \frac{y_t}{\hat{y}_t} - 1 \right] \quad (35)$$

D.2. Calibration quality metrics

Empirical coverage rate (ECR)

$$\text{ECR}_\alpha = \frac{1}{N} \sum_{t=1}^N \mathbf{1}\{y_t \in [\hat{y}_t^{\text{lower}}, \hat{y}_t^{\text{upper}}]\} \quad (36)$$

Coverage deviation (CD)

$$\text{CD} = |\text{ECR} - (1 - \alpha)| \times 100\% \quad (37)$$

Interval width quality (IWQ)

$$\text{IWQ} = \frac{\text{Std}(\hat{y}_t^{\text{upper}} - \hat{y}_t^{\text{lower}})}{\text{Mean}(\hat{y}_t^{\text{upper}} - \hat{y}_t^{\text{lower}})} \quad (38)$$

Higher IWQ indicates greater adaptive capacity.

Kupiec's unconditional coverage test (?)

$$LR_{\text{uc}} = -2 \log \left[\frac{(1 - \alpha)^{N-V} \alpha^V}{(1 - \hat{p})^{N-V} \hat{p}^V} \right] \sim \chi_1^2 \quad (39)$$

where V is the number of violations and $\hat{p} = V/N$. Reject null if $LR_{\text{uc}} > 3.841$ ($p < 0.05$).

D.3. Economic Value Metrics

Portfolio variance reduction (PVR)

$$\text{PVR} = \frac{\text{Var}[\mathbf{r}_{\text{baseline}}] - \text{Var}[\mathbf{r}_{\text{RADE-UQ}}]}{\text{Var}[\mathbf{r}_{\text{baseline}}]} \times 100\% \quad (40)$$

VaR hit rate

$$\text{Hit rate}_\alpha = \frac{1}{N} \sum_{t=1}^N \mathbf{1}\{r_t < -\text{VaR}_{\alpha,t}\} \quad (41)$$

Sharpe ratio

$$\text{Sharpe} = \frac{\bar{r} - r_f}{\sigma_r} \quad (42)$$

Realized utility gain (?)

$$\Delta U = \left(\bar{r}_{\text{RADE}} - \frac{\gamma}{2} \sigma_{\text{RADE}}^2 \right) - \left(\bar{r}_{\text{base}} - \frac{\gamma}{2} \sigma_{\text{base}}^2 \right) \quad (43)$$

Where γ is the risk aversion parameter (typically 3-5)?

E. Software and reproducibility

E.1. Software stack

Core libraries

- Python 3.9.12
- PyTorch 2.0.1 (RDN implementation)
- LightGBM 3.3.5
- scikit-learn 1.3.0 (random forest, HAR-RV)
- pandas 1.5.3, NumPy 1.24.3
- matplotlib 3.7.1, seaborn 0.12.2

Hardware

- CPU: Intel Xeon E5-2680 v4 @ 2.40GHz (28 cores)
- RAM: 64GB DDR4
- OS: Ubuntu 20.04 LTS

E.2. Directory structure

RADE-UQ/

data/

raw/

Raw data files

processed/

Preprocessed features

splits/

Train/val/test splits

models/

har_rv.py

HAR-RV implementation

```

    random_forest.py      # Random Forest + bootstrap
    lightgbm_model.py     # LightGBM + bootstrap
    rdn.py                # Regime Detection Network
ensemble/
    weighting.py          # Triple-adaptive weighting
    uncertainty.py        # UQ decomposition
    calibration.py        # Post-hoc calibration
evaluation/
    metrics.py            # All 18 evaluation metrics
    visualization.py      # Plotting functions
experiments/
    main.py               # Main training script
    ablation.py           # Ablation studies
    robustness.py         # Robustness checks
notebooks/
    data_exploration.ipynb
    results_analysis.ipynb
    figures_generation.ipynb
requirements.txt          # Python dependencies
README.md                # Documentation

```

E.3. Execution Instructions

Step 1: Install dependencies

```
pip install -r requirements.txt
```

Step 2: Download and preprocess data

```
python data/download_data.py
python data/preprocess.py
```

Step 3: Train models

```
python experiments/main.py --config configs/default.yaml
```

Step 4: Evaluate and generate results

```
python experiments/evaluate.py --checkpoint models/rade_uq_best.pt
```

Step 5: Generate figures

```
jupyter notebook notebooks/figures_generation.ipynb
```

The complete codebase, including asset-specific notebooks for all five markets (S&P 500, NASDAQ-100, Russell 2000, EUR/USD, Gold), is publicly available at: <https://github.com/JeaabKhalid/RADE-UQ-Regime-Adaptive-Deep-Ensemble-with-Uncertainty-Quantification-.git>.

The repository includes all modules described in Appendix E.3, requirements files, and a `README.md` with step-by-step setup instructions.

Total execution time: ≈ 50 hours (training) + 2 hours (evaluation/visualization).



AIMS Press

© 2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0>)