

7507178-forecasting-realized-volatility-using-idiosyncratic-and-systematic-jump-risks-a-

SUMMARY: The article investigates the relative predictive role of idiosyncratic and systematic jump risks in forecasting realized volatility across major U.S. equity market sectors. The study compares linear benchmark models with a broad set of machine learning methods, finding that selected machine learning models improve forecasting accuracy relative to the High-Frequency ARCH (HAR) model. Feature importance analysis indicates that systematic jump risk contributes more strongly to volatility predictability than idiosyncratic jump risk. The authors propose novel regime-dependent predictors derived from a Markov regime-switching framework and incorporate them into the learning process, leading to a marked improvement in forecasting performance, particularly for volatility direction prediction. The findings suggest that integrating jump-based feature engineering and regime-dependent information within machine learning frameworks yields more accurate and robust volatility forecasts.

METHOD: The study conducts a comprehensive out-of-sample comparison of linear benchmark models and a broad set of machine learning methods. Feature importance analysis is used to assess the relative predictive importance of systematic co-jump and idiosyncratic-jump measures in volatility forecasting. Novel regime-dependent predictors derived from a Markov regime-switching framework are incorporated into the learning process.

KEY FINDINGS:

- Machine learning models improve forecasting accuracy relative to the HAR model.
- Systematic jump risk contributes more strongly to volatility predictability than idiosyncratic jump risk.
- Novel regime-dependent predictors lead to improved forecasting performance.
- The integration of jump-based feature engineering and regime-dependent information yields more accurate and robust volatility forecasts.

IMPLICATION FOR TRADING: The findings suggest that practitioners can improve their volatility forecasting models by incorporating jump-based features and regime-dependent information into machine learning frameworks. This can lead to more accurate and robust forecasts, which are essential for risk management, portfolio allocation, and derivative pricing in financial markets. By understanding the distinct sources of risk and their predictive effects, traders can make more informed decisions and potentially enhance their trading strategies.

Forecasting Realized Volatility Using Idiosyncratic and Systematic Jump Risks: A Machine Learning Approach with Market Regimes

Weijia Peng^{a,*}, Eman Abdelfattah^b

^a*Department of Finance, Jack Welch College of Business & Technology, Sacred Heart University, USA*

^b*School of Computing & Engineering, College of Arts & Sciences, Sacred Heart University, USA*

Abstract

Accurate forecasting of equity market volatility is essential for risk management, portfolio allocation, and derivative pricing. From a machine learning and soft-computing perspective, this paper investigates the relative predictive role of idiosyncratic and systematic jump risks in forecasting realized volatility across major U.S. equity market sectors, considering both volatility levels and directional movements. We conduct a comprehensive out-of-sample comparison of linear benchmark models and a broad set of machine learning methods. The results show that selected machine-learning models improve forecasting accuracy relative to HAR, but the gains depend on the sector and algorithm, and HAR remains a competitive benchmark. Feature importance analysis indicates that systematic jump risk contributes more strongly to volatility predictability than idiosyncratic jump risk. To further capture nonlinear and time-varying market dynamics, we propose novel regime-dependent predictors derived from a Markov regime-switching framework and incorporate them into the learning process. These regime-sensitive features lead to a marked improvement in forecasting performance, particularly for volatility direction prediction. Overall, the findings demonstrate that integrating jump-based feature engineering and regime-dependent information within machine learning frameworks yields more accurate and robust volatility forecasts.

Keywords: Volatility Forecasting, Idiosyncratic and Systematic Risk, Machine Learning, Markov regime-switching.

*Corresponding author.

Email addresses: pengw@sacredheart.edu (Weijia Peng), abdelfattah@sacredheart.edu (Eman Abdelfattah)

1. Introduction

Accurately forecasting volatility in financial markets remains a fundamental problem with important implications for risk management, trading strategies, and portfolio optimization. The realized volatility of the S&P 500 index has attracted substantial attention from both academics and practitioners due to its central role in global financial markets. Movements in the S&P 500 not only reflect U.S. market conditions but also generate significant spillover effects across international markets [1]. Within the index, strong sectoral interconnections—such as those among financials, technology, energy, and utilities—may further amplify volatility transmission. Shocks originating in one sector can propagate to others, underscoring the interconnected nature of modern financial systems and the need for accurate volatility forecasting at the sectoral level.

Volatility forecasting has evolved from conditional heteroskedasticity models such as ARCH and GARCH [2, 3] to realized-volatility models that exploit information from high-frequency data. Among these approaches, the HAR model [4] has become a standard forecasting benchmark because it parsimoniously captures volatility persistence across daily, weekly, and monthly horizons. However, its linear specification may not fully accommodate nonlinear relationships, predictor interactions, or structural changes in volatility dynamics. Machine-learning methods provide flexible alternatives that can capture nonlinearities and interactions within expanded predictor sets. Recent studies have applied neural networks, penalized regression, tree-based algorithms, and ensemble methods to realized-volatility forecasting, with evidence of improvements over conventional econometric benchmarks in some settings [5–9]. Nevertheless, challenges remain regarding model interpretation and the identification of economically meaningful predictors.

Within the volatility-forecasting literature, jump dynamics are recognized as an important source of market fluctuations. [10] incorporated continuous and aggregate jump components into a linear HAR forecasting framework, while [11] further decomposed sector-level jump variation according to its economic source. Idiosyncratic jumps (IJ) capture shocks affecting an individual sector in isolation, whereas systematic co-jumps (CJ) capture shocks shared simultaneously across sectors. These two components therefore represent economically distinct sources of risk and may have different predictive effects on future volatility. However, these studies do not systematically examine the systematic co-jump and idiosyncratic-jump decomposition within an ML framework. Building on [10] and [11], the present study extends the established RV–CJ–IJ predictor structure from linear HAR models to a broad set of ML algorithms. Thus, our contribution is not the initial construction of the CJ/IJ measures, but their systematic evaluation within nonlinear ML models. We also use feature-importance analysis to assess the relative predictive importance of systematic co-jump and idiosyncratic-jump measures in volatility forecasting. By holding the RV–CJ–IJ information set fixed across the linear and ML specifications, we examine whether flexible learning algorithms can capture nonlinearities among realized volatility, systematic co-jump risk, and idiosyncratic jump risk that are not accommodated by the linear HAR model. The results show that selected ML methods deliver more consistent forecasting gains than the linear specification across the sectoral series.

From a soft-computing perspective, this study focuses on enhancing machine-learning volatility fore-

casting through structured feature engineering, combining high-frequency jump measures and regime-dependent information to improve predictive performance and interpretability. The objective of this paper is to forecast the realized volatility of major S&P 500 sectors by incorporating measures of idiosyncratic and systematic jump risks within a machine learning framework. Unlike much of the existing literature, which relies primarily on linear benchmark models such as HAR, we employ a broad set of machine learning methods to capture potential nonlinearities and complex interactions. Specifically, we consider decision trees, random forests, extremely randomized trees, AdaBoost, stochastic gradient descent (SGD), extreme gradient boosting (XGBoost), k-nearest neighbors (KNN), and multilayer perceptron (MLP) neural networks.

In addition to point forecasts of realized volatility, we examine the prediction of volatility direction, which is particularly relevant for practical applications such as trading strategies and risk management. Directional forecasting performance is evaluated using classification-based machine learning models, including decision tree, random forest, extra-trees, AdaBoost, SGD, KNN, and MLP classifiers. Feature importance analyses are conducted to assess the relative contributions of lagged realized volatility, idiosyncratic jump risks, and systematic co-jump risks.

A further contribution of this study lies in the construction of regime-dependent predictors derived from Markov regime-switching models. While prior studies have primarily used regime-switching models as standalone forecasting tools [12–19], we adopt a different approach. We first identify high- and low-volatility regimes and then extract regime-specific features that are incorporated into the machine learning forecasting framework. This allows us to capture nonlinear structural changes in market conditions while maintaining the flexibility of ML-based models.

The empirical analysis focuses on one-day-ahead forecasts of realized volatility over the period from February 6, 2007, to July 29, 2026, covering 4,899 trading days. Realized volatility is constructed from 5-minute intraday returns. The analysis covers major S&P 500 sectors, including Financials (XLF), Technology (XLK), Utilities (XLU), Consumer Staples (XLP), and Consumer Discretionary (XLY). Co-jumps and idiosyncratic jumps are identified following [8, 11, 20, 21]. Forecast performance is evaluated using out-of-sample R^2 , mean squared error (MSE), and mean absolute error (MAE), while directional forecasts are assessed using classification accuracy, confusion matrices, and the F_1 -score.

The relative forecasting performance of machine-learning models and HAR varies across sectors and algorithms. Selected machine-learning models improve upon HAR in several sectors, while HAR remains competitive and achieves the lowest test MSE for XLU. These findings support the potential value of machine-learning methods without implying a universal advantage over the linear benchmark. Among the ML approaches, XGBoost delivers particularly strong and stable performance, achieving higher out-of-sample R^2 values while maintaining relatively short training times. Feature importance analysis indicates that lagged realized volatility is the most informative predictor, with systematic jump risks typically ranking ahead of idiosyncratic jumps across sectors. Moreover, parsimonious models using only a small subset of key predictors—lagged realized volatility and systematic jump risk, and in some cases idiosyncratic jump risk—achieve accuracy comparable to models using a larger feature set. Regime-switching-based features provide additional gains primarily in directional forecasting tasks, improving

classification accuracy and F_1 -scores.

From a machine-learning and soft-computing perspective, this study makes four main contributions. First, we embed jump-based volatility measures within a broad machine learning forecasting framework and systematically compare their predictive performance with the benchmark Heterogeneous Autoregressive (HAR) model. Second, we provide a comprehensive empirical assessment of the relative importance of idiosyncratic and systematic jump risks for sector-level volatility forecasting. Third, we extend volatility forecasting beyond point predictions by jointly examining directional changes in realized volatility using classification-based ML models. Fourth, we introduce regime-dependent features derived from Markov regime-switching models and integrate them into machine learning algorithms, demonstrating that such features are particularly valuable for improving directional volatility forecasts.

The remainder of this paper proceeds as follows. Section II reviews the related literature. Section III outlines the research methodology. Section IV describes the data and sample. Section V presents the empirical results. Section VI concludes. Section VII discusses the limitations and directions for future research.

2. Literature Review

Our research lies in three strands of the finance literature. The first strand is volatility forecasting. Early approaches relied on econometric models such as ARCH [2] and GARCH [3], which successfully captured volatility clustering. Extensions like EGARCH [22] and GJR-GARCH [23] improved the treatment of asymmetry and leverage effects. In parallel, stochastic volatility models [24] modeled volatility as an unobserved process, offering greater flexibility but at higher computational cost. The introduction of high-frequency trading data led to realized volatility [25] and the HAR-RV model [4, 26], which provided accurate short-horizon forecasts and better long-memory features. In recent years, research has expanded toward machine learning approaches. Machine learning methods, such as deep learning and tree-based models or ensemble and hybrid models [6, 27], have shown promise in handling nonlinearity and large datasets, often outperforming traditional models in empirical applications. Recent empirical evidence strengthens this trend: investor-confidence-based machine learning models have been shown to improve realized volatility forecastability [7]; high-frequency neural network and penalized regression models achieve state-of-the-art accuracy in stock-level volatility prediction [6]; and hybrid decomposition machine learning frameworks deliver substantial gains in forecasting equity index volatility [28]. [29] propose a shrinkage HAR framework with high-dimensional global predictors that shows LASSO-type shrinkage delivers robust gains in volatility, directional, density, and asset-allocation forecasting across markets and horizons. However, no single approach dominates across all horizons and contexts [30]. Classical econometric models remain valued for interpretability, while AI-based methods push the frontier in predictive accuracy, suggesting that future research will likely emphasize hybrid, explainable, and robust volatility forecasting frameworks.

An important complementary strand of the volatility forecasting literature highlights the role of regime-switching models. Early applications showed that realized volatility forecasts could be improved by conditioning on regime dynamics, especially during crises and market turbulence. For instance,

[13] introduce a Markov regime-switching model to the heterogeneous autoregressive model to forecast realized volatility of the crude oil futures market. [18] adopt Markov switching analysis to uncover regime switching of the INE crude oil futures market and forecast the realized volatility of INE crude oil futures. [17] construct a set of HAR models with three types of infinite Hidden Markov regime-switching structures to forecast five agricultural commodity futures (corn, cotton, indica rice, palm oil, and soybeans) based on high-frequency data from Chinese futures markets. Related extensions incorporating regime-switching models include [14], [15], [16], and [19].

The third strand of literature is about jumps and co-jumps and their tests. The analysis of jumps and co-jumps in asset prices has become a central topic in financial econometrics, particularly with the rise of high-frequency data. Early contributions, such as [31], developed realized power variation and bipower variation methods to statistically distinguish continuous volatility from jumps. Extensions by [20] and [32] introduced test statistics capable of identifying jumps at specific time points, allowing for event-level inference.

Building on these methods, researchers turned to co-jumps, which occur when multiple assets experience simultaneous jumps, often reflecting systemic shocks or macroeconomic news. [33] proposed statistical procedures for testing co-jumps, while [34] linked co-jumps to risk premia, showing that joint discontinuities carry valuable pricing information. [35] introduced realized power and bipower variation, providing a nonparametric approach for separating the continuous and jump components of price variation. Subsequent studies developed tests for identifying jumps at specific points in time [20, 31, 32]. Similarly, [36] highlight how high-frequency measures improve the identification of common jump components across assets. The literature also connects jumps and co-jumps to macroeconomic announcements and financial crises. [10] documented how scheduled news releases generate predictable jump patterns in foreign exchange and bond markets. More recently, [37] and [38] developed methods to link jump risk to stochastic volatility and tail risk pricing. Table 1 shows the comparison of our work with other related works.

3. Research Methodology

3.1. Realized volatility measures

We follow the realized-volatility framework and notation of [39]. Daily realized volatility is constructed from 5-minute intraday log returns as

$$\widehat{RV}_t = \sum_{j=1}^m (\Delta X_{j,t})^2. \quad (1)$$

where $\Delta X_{j,t}$ denotes the j -th intraday return on day t and $m = 78$ is the number of 5-minute intervals in a trading day. Realized volatility estimates the quadratic variation of the log-price process and therefore reflects both its continuous and jump components. We use bipower variation to estimate the continuous component and construct the jump measures used in the forecasting analysis. The underlying semimartingale framework, quadratic-variation decomposition, and formal definition of bipower variation are provided in Section S.1 of the Online Supplemental Appendix.

Table 1: COMPARISON OF THIS WORK WITH OTHER RELATED WORKS

Article	Dataset	Best Reported Performance	Task
Gupta et al. (2023)	Daily price data of the DJIA, 2001–2020	Relative MSE as low as 0.947	Forecast monthly realized volatility
Zhang et al. (2025)	S&P 500 5-minute intraday data, 2015–2023	CEEMDAN–RF hybrid reduces MSE by 56.66% relative to baseline model	Forecast daily realized volatility
Christensen et al. (2023)	29 DJIA stocks 5-minute intraday data, 2001–2017	Neural networks and elastic net achieve relative MSE of about 0.88–0.90	Forecast daily realized volatility
<i>Our Study</i>	Major S&P 500 sector ETFs 5-minute intraday data, 2007–2026	For XLY, best model achieves out-of-sample $MSE = 0.744$	Forecast daily realized volatility

3.2. Detection of joint and idiosyncratic jumps

To distinguish common jumps from asset-specific jumps across a panel of M assets, we employ a co-exceedance rule based on the jump tests proposed in [20] and the combined LM–BNS framework discussed in [21]. Our implementation follows the general procedure in [36]. Let $\mathcal{J}_s^{(k)}$ denote an indicator equal to one when a jump is detected in asset k during the intraday interval containing time s , and zero otherwise. The co-jump decision rule is then

$$\sum_{k=1}^M \mathcal{J}_s^{(k)} \begin{cases} \geq 2, & \text{classified as a co-jump,} \\ \leq 1, & \text{no co-jump.} \end{cases} \quad (2)$$

Let the trading day be partitioned into m intraday intervals of length δ , and let $\Delta X_{j,t}$ denote the j -th intraday return on day t . Using the co-jump indicator denoted $\mathcal{C}_{j,t}$, we define the contribution of co-jumps to daily return variation as

$$CJ_t = \sum_{j=1}^m (\Delta X_{j,t})^2 \mathcal{C}_{j,t}. \quad (3)$$

The indicator $\mathcal{C}_{j,t} = 1$ when the target asset experiences a jump during interval j on day t and the rule in (2) identifies a co-jump at that interval; otherwise $\mathcal{C}_{j,t} = 0$.

Similarly, idiosyncratic (asset-specific) jump variation is computed using the indicator $\mathcal{I}_{j,t}$, which flags intervals where only the target asset exhibits a jump:

$$IJ_t = \sum_{j=1}^m (\Delta X_{j,t})^2 \mathcal{I}_{j,t}. \quad (4)$$

Here, $\mathcal{I}_{j,t} = 1$ when a jump occurs in the target asset but the co-jump condition in (2) is not met. In the empirical analysis, we set the number of intraday observations per day to $m = 78$.

3.3. HAR Model

In this section, we outline the models employed to forecast sector-level equity volatility. The HAR-RV model, originally introduced by [4], builds upon the GARCH framework and the Heterogeneous Market Hypothesis. This hypothesis suggests that market participants operate over different investment horizons, leading to heterogeneous effects on volatility. Accordingly, the HAR-RV specification incorporates realized volatility measures across multiple time horizons. A baseline version of the HAR-RV model can be expressed as

$$RV_{t+h} = \beta_0 + \beta_d RV_t + \beta_w RV_{t-5,t} + \beta_m RV_{t-22,t} + \epsilon_{t+h}, \quad (5)$$

where RV_{t+h} is the realized volatility for day $t+h$. $RV_{t-h,t}$ is the average of realized volatility values over the given horizon h , i.e.,

$$RV_{t-h,t} = h^{-1} [RV_{t-h+1} + RV_{t-h+2} + \dots + RV_t]. \quad (6)$$

Extending the HAR-RV framework, [10] decompose realized variation into its continuous and jump components, yielding the HAR-RV-CJ model. To assess the role of co-jumps and idiosyncratic jumps in predicting future volatility, we employ a set of HAR-RV-C-CJ-IJ model variants that explicitly incorporate the jump components (CJ_t) and (IJ_t):

$$\begin{aligned} \log RV_{t+1} = & \beta_0 + \beta_{cd} \log C_t + \beta_{cw} \log C_{t-5,t} \\ & + \beta_{cm} \log C_{t-22,t} + \beta_{cjd} \log CJ_t \\ & + \beta_{cjl} \log CJ_{t-5,t} + \beta_{cjm} \log CJ_{t-22,t} \\ & + \beta_{ijd} \log IJ_t + \beta_{ijw} \log IJ_{t-5,t} \\ & + \beta_{ijm} \log IJ_{t-22,t} + \epsilon_{t+1} \end{aligned} \quad (7)$$

where C_t represents the continuous variation, estimated by the bipower variation (BPV). The right-hand-side variables in the equation include the daily, weekly, and monthly lag terms.

3.4. Machine Learning Models

We evaluate a broad set of machine learning methods to capture linear and nonlinear relationships between realized volatility and the jump-based predictors. The regression models include Decision Tree, Random Forest, Extra Trees, AdaBoost, SGD, XGBoost, KNN, and MLP. For directional forecasting, we additionally consider Logistic Regression, Gaussian Naive Bayes, SVM, CatBoost, and several stacking ensembles. These models represent linear, distance-based, tree-based, boosting, neural-network, and ensemble approaches, allowing us to compare predictive performance across model classes with different levels of flexibility and interpretability. Detailed descriptions of the individual algorithms and their implementation are provided in Section S.2 of the Online Supplemental Appendix.

To assess the sensitivity of the classification results to the training-test cutoff, we supplement the baseline chronological 80:20 split with a chronological 70:30 split. In each case, the earlier observations constitute the training sample and the remaining observations form the test sample. For the regime-augmented CatBoost model, test F1 scores under the 80:20 and 70:30 splits are, respectively, 0.7188 and

0.7127 for XLY, 0.7367 and 0.7402 for XLU, 0.6989 and 0.7116 for XLK, 0.7326 and 0.7374 for XLP, and 0.7184 and 0.7041 for XLF. These results indicate relatively similar performance for this model across the two chronological cutoffs.

We also report the mean and standard deviation of F1 scores across three stratified cross-validation folds within the training sample, implemented using `StratifiedKFold(n_splits=3, shuffle=True, random_state=42)`. These statistics summarize performance variation across validation subsets and are reported separately from test-sample metrics. Because the folds are shuffled, they provide supplementary evidence on consistency across subsets rather than stability across chronological periods. Detailed results are provided in Section S.7 of the Online Supplemental Appendix. Table S.18 compares test accuracy and F1 scores under the chronological 80:20 and 70:30 splits, while Table S.19 reports the mean and standard deviation of F1 scores across three training-sample cross-validation folds.

3.5. Markov Regime-Switching Models

To capture changes in market conditions, we employ a two-state Markov regime-switching model that classifies each period as either a low- or high-volatility state based on observed realized volatility [16]. The model produces filtered regime probabilities using information available through time t . We use the estimated probability of the high-volatility state as a regime-dependent predictor in the subsequent regression and classification models. During the test period, regime probabilities are generated recursively without using future information, thereby avoiding look-ahead bias.

As a robustness check, we also estimate a three-state Markov regime-switching model that distinguishes among low-, medium-, and high-volatility conditions. This alternative specification provides a more granular representation of volatility dynamics and allows us to assess whether the predictive contribution of regime-dependent information is sensitive to the assumed number of latent states. Model parameters are estimated exclusively using the training sample and held fixed during test-period filtering. Both specifications are evaluated and their out-of-sample results are reported. The stronger performance of the three-state specification in the reported comparison is therefore an empirical finding for the evaluation sample, rather than evidence that three states were selected independently of test-sample performance or constitute a generally optimal specification. The complete specifications, filtering recursions, estimation procedures, and results for the two- and three-state models are provided in Section S.3 of the Online Supplemental Appendix.

3.6. Feature Construction and Forecasting Procedure

To ensure reproducibility and prevent information leakage, all features and models are constructed using the following procedure. Lagged realized volatility, idiosyncratic jump variation, and systematic co-jump variation are calculated using information available through day t and are used to forecast realized volatility or its direction on day $t + 1$. The Markov-switching models are estimated using the training sample only. Test-period regime probabilities are generated recursively using the fixed training-sample parameters and the Hamilton filter, without using future observations. Regime labels are ordered according to their estimated mean realized volatility and identified as low and high in the

two-state model and low, medium, and high in the three-state model. StandardScaler was used before applying all models to ensure that each feature in the dataset is standardized before being passed to each model. Hyperparameters are selected using three-fold stratified cross-validation, implemented with `StratifiedKFold(n_splits=3, shuffle=True, random_state=42)`. Cross-validation is conducted exclusively on `X_train` and `y_train`, and the selected hyperparameters are subsequently held fixed for test-sample evaluation. Although this procedure excludes the final test sample from hyperparameter selection, shuffling does not preserve chronological ordering within the validation folds. Consequently, the tuning procedure does not employ rolling- or expanding-window validation and may not fully reflect the temporal structure of the forecasting task. We acknowledge this as a limitation. Sector-specific hyperparameter values are reported in Table S.17 of the Online Supplemental Appendix.

4. Data and Sample

We utilized intraday data for five SPDR exchange-traded funds (ETFs), each representing a distinct sector of the S&P 500 index: financials (XLF), technology (XLK), consumer staples (XLP), utilities (XLU), and consumer discretionary (XLY). The high-frequency trade and quote (TAQ) database, accessed via the Wharton Research Data Services (WRDS), served as the source of these observations. The sample period spans from February 6, 2007, to July 29, 2026, covering 4,899 trading days. To maintain consistency, overnight returns were excluded. Price filtering relied on the previous-tick method of [40], while data cleaning and outlier removal followed the approach of [41]. To mitigate microstructure noise, returns were sampled at 5-minute intervals, typically yielding 78 intraday observations per trading day.

Missing intraday prices were forward-filled from the last available observation. Following the cleaning approach of [41], absolute one-minute log returns exceeding a fixed threshold of 0.005 for the sector ETFs were flagged as potential outliers and further evaluated using local sign-reversing jump-back patterns and a same-day volatility reference. The fixed threshold of 0.005 was adopted from the data-cleaning procedure of [41]. These data-dependent checks used information available by the end of day t , consistent with constructing daily predictors for forecasting day $t + 1$. Flagged log prices meeting the correction criteria were replaced by the average of the surrounding endpoint log prices. This procedure aims to reduce artificial inflation of realized volatility and spurious jump detections caused by recording errors. However, it may also smooth genuine short-lived price movements and affect the magnitude or detection of jumps and co-jumps. The resulting measures are therefore conditional on the preprocessing choices.

5. Empirical results

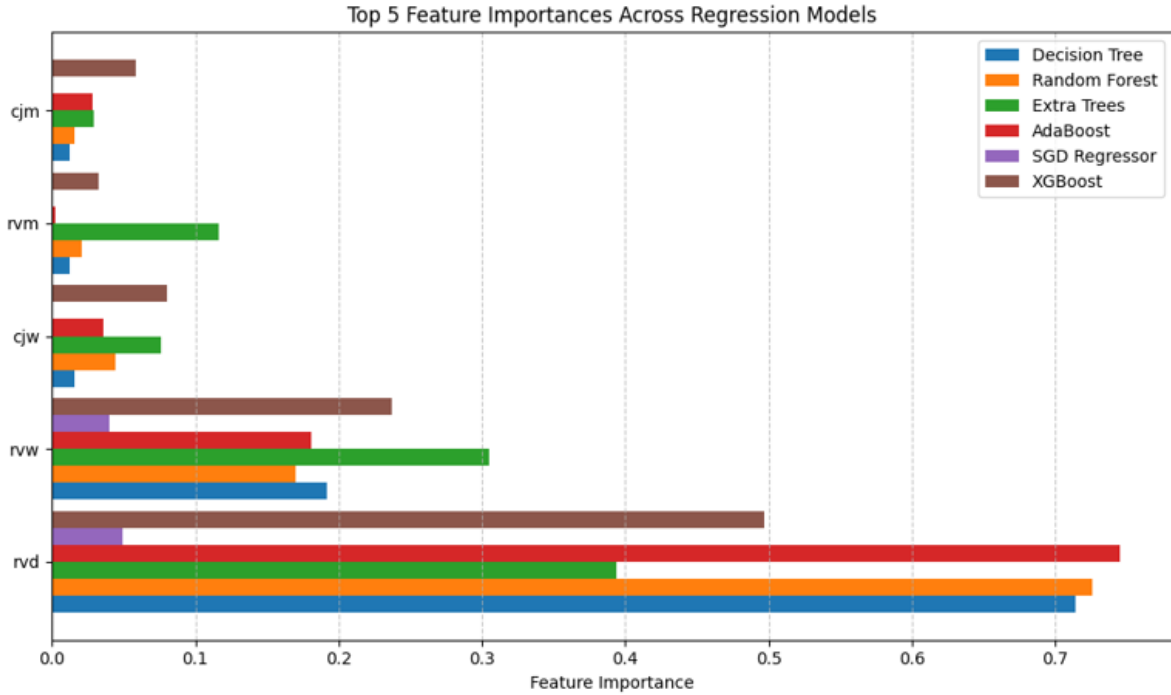
Table 2 summarizes the prediction targets, comprising five major sectoral exchange-traded funds (ETFs) within the S&P 500: Financials (XLF), Technology (XLK), Consumer Staples (XLP), Utilities (XLU), and Consumer Discretionary (XLY). Table 3 reports descriptive statistics, including the mean, standard deviation, skewness, and kurtosis, for key variables: realized volatility (RV), bi-power variation (BPV), co-jumps (CJ), and idiosyncratic jumps (IJ) across these sectors. The results indicate that realized volatility exhibits higher mean and standard deviation values compared with other measures.

Similarly, co-jump variation shows greater mean and standard deviation than idiosyncratic jump variation, and displays higher skewness and kurtosis relative to the other three measures. Because the main findings are qualitatively consistent across sectors, we discuss the forecasting results for the Consumer Discretionary sector (XLY) in detail in the main text. Complete results for the remaining sectors are reported in Sections S.4–S.6 of the Online Supplemental Appendix.

Table 2: Forecasting Target Variables

Target	Description	Transformation	Freq
XLF	Financial Sector SPDR Fund	$\ln(X_t) - \ln(X_{t-1})$	Daily
XLK	Technology Sector SPDR Fund	$\ln(X_t) - \ln(X_{t-1})$	Daily
XLP	Consumer Staples SPDR Fund	$\ln(X_t) - \ln(X_{t-1})$	Daily
XLU	Utilities SPDR Fund	$\ln(X_t) - \ln(X_{t-1})$	Daily
XLY	Consumer Discretionary SPDR Fund	$\ln(X_t) - \ln(X_{t-1})$	Daily

Notes: This table reports the prediction targets. Data transformations used in forecasting experiments are in the third column. See Section 3 for details.



Notes: Feature importances are normalized within each model. *rvd*, *rvw*, and *rvm* denote the daily, weekly, and monthly lags of realized volatility, respectively. *ijd*, *ijw*, and *ijm* represent the daily, weekly, and monthly lags of idiosyncratic jump variation. *cjd*, *cjw*, and *cjm* denote the daily, weekly, and monthly lags of co-jump variation.

Figure 1: Feature-Importance Results Across Regression Models.

These forecasting experiments were conducted in three phases. In Phase I, eight machine learning regressors were utilized as our target variables are continuous numeric variables. The dataset was parti-

Table 3: Descriptive Statistics for Sector ETFs

Target	Measure	Mean	Std	Skew	Kurt
XLY	RV	0.151	0.103	3.282	21.144
	BPV	0.145	0.098	3.091	18.712
	CJ	0.040	0.092	4.724	35.883
	IJ	0.010	0.029	3.259	15.152
XLU	RV	0.150	0.092	5.314	47.261
	BPV	0.143	0.088	5.193	44.453
	CJ	0.037	0.090	6.130	60.465
	IJ	0.019	0.039	2.436	11.255
XLK	RV	0.153	0.098	3.467	23.654
	BPV	0.147	0.095	3.303	20.881
	CJ	0.037	0.090	5.109	41.257
	IJ	0.011	0.031	3.825	28.073
XLP	RV	0.111	0.070	5.225	48.208
	BPV	0.104	0.066	5.216	49.432
	CJ	0.031	0.069	5.988	59.770
	IJ	0.010	0.023	2.805	13.137
XLF	RV	0.174	0.142	3.966	26.510
	BPV	0.168	0.139	3.879	25.345
	CJ	0.044	0.110	6.043	61.027
	IJ	0.017	0.052	4.958	38.606
<i>Note:</i> XLY, XLU, XLK, XLP, and XLF each contain $N = 4,899$ observations. RV denotes realized volatility; BPV denotes bipower variation; CJ and IJ represent co-jump and idiosyncratic jump components, respectively. Further details on variable construction Refer to Section 3.					

tioned into an 80:20 ratio for training and testing. Grid search was conducted to build high-performing, reliable models along with 3-fold cross-validation to evaluate how well models will perform on unseen data. Table 4 presents the results obtained. Extra Trees and XGBoost jointly achieve the highest Test R^2 (0.744), with Extra Trees attaining the lowest Test MSE (0.00247) and XGBoost the lowest Test MAE (0.0308).

Table 5 reports model-specific feature-importance measures for XLY, and Figure 1 illustrates selected prominent predictors. For Decision Tree, importance is based on normalized, sample-weighted reductions in impurity; Random Forest and Extra Trees aggregate these measures across trees. AdaBoost uses a weighted aggregation of its base estimators' importances, while tree-based XGBoost uses normalized gain-based importance under its default setting. Because these measures differ across algorithms, we interpret feature rankings within each model rather than compare their numerical magnitudes directly across models.

The daily realized-volatility lag (rvd) ranks highest in each reported XLY model, and the weekly

Table 4: Regression Model Performance Comparison – Prediction Target XLY

Model	CV R^2	Test R^2	Test MSE	Test MAE	Training Time (s)
HAR (Linear Model)	–	0.737	0.00254	0.0310	0.003
Decision tree	0.601	0.652	0.00336	0.0345	0.460
Random forest	0.727	0.724	0.00266	0.0315	8.620
Extra trees	0.734	0.744	0.00247	0.0314	1.290
AdaBoost	0.718	0.698	0.00292	0.0341	8.019
SGD	0.737	0.732	0.00259	0.0313	27.260
XGBoost	0.733	0.744	0.00248	0.0308	2.959
KNN	0.706	0.722	0.00268	0.0321	1.344
MLP	0.675	0.659	0.00329	0.0346	10.803
<i>Notes:</i> This table reports the out-of-sample performance of the Heterogeneous Autoregressive (HAR) model and various machine learning models in forecasting the realized volatility of the Consumer Discretionary sector (XLY). CV R^2 denotes the cross-validated coefficient of determination, and Test R^2 denotes the out-of-sample fit on the test set. The final column reports average training time in seconds. Boldface values indicate the best performance within each metric.					

Table 5: Feature Importances Across ML Models

Feature	Decision Tree	Random Forest	Extra Trees	Ada Boost	SGD Re- gressor	XG Boost
rvd	0.714	0.726	0.394	0.745	0.049	0.497
rvw	0.192	0.170	0.305	0.181	0.040	0.237
rvm	0.012	0.020	0.116	0.003	0.000	0.032
ijd	0.008	0.000	0.001	0.000	0.000	0.014
ijw	0.018	0.004	0.005	0.001	0.006	0.022
ijm	0.013	0.007	0.006	0.001	0.004	0.027
cjd	0.014	0.013	0.067	0.006	0.003	0.031
cjw	0.015	0.044	0.076	0.036	0.000	0.080
cjm	0.013	0.016	0.029	0.029	0.000	0.059
<i>Notes:</i> The table reports feature importance values across regression models used to forecast realized volatility. rvd, rvw, and rvm denote the daily, weekly, and monthly lags of realized volatility, respectively. ijd, ijw, and ijm represent the daily, weekly, and monthly lags of idiosyncratic jump variation. cjd, cjw, and cjm denote the daily, weekly, and monthly lags of co-jump variation, respectively. Boldface values indicate the highest feature importance for each model.						

lag (rvw) also ranks prominently. Monthly realized volatility and selected co-jump predictors receive additional importance, with their relative rankings varying across algorithms. These patterns indicate that the fitted models rely primarily on recent realized volatility, while jump-based predictors provide additional information whose importance is model-dependent. The supplementary sector-level results similarly highlight the prominence of lagged realized volatility.

In Phases II and III, seventeen classification models were utilized. Twelve of these models used in these phases were individual classifiers, and the remaining five models were stacked ensemble classifiers. The dataset was partitioned into an 80:20 ratio for training and testing. Also, in these two phases grid search was conducted along with 3-fold cross-validation.

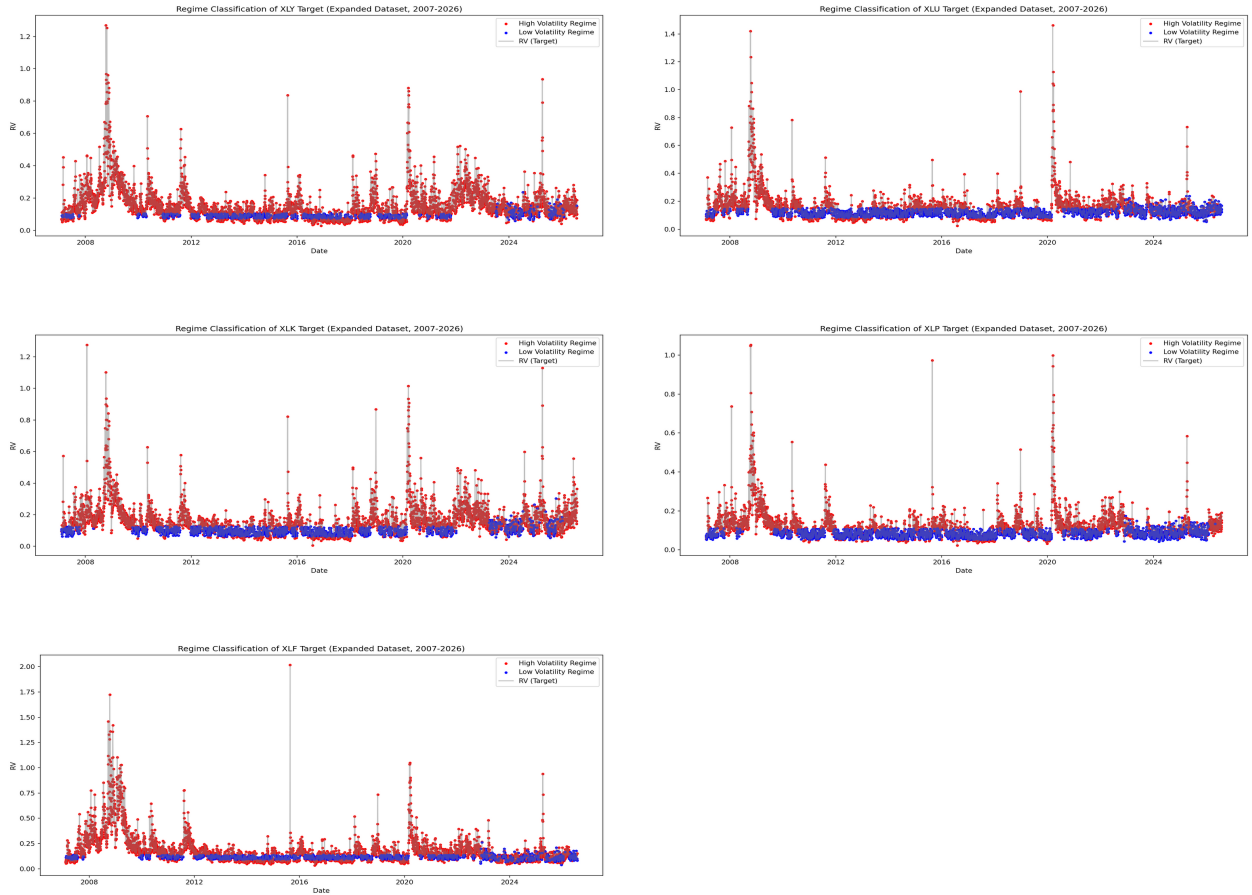
Before proceeding to Phase II, we incorporated a Markov regime-switching model to capture latent volatility dynamics and classify market conditions into distinct regimes. To assess the sensitivity of the regime specification, we estimate both two- and three-state Markov-switching models using the 2007–2026 sample. Figures 2 and 3 show that both specifications identify major stress episodes, particularly the 2008–2009 global financial crisis and the COVID-19 shock in 2020, as high-volatility periods across sectors. However, the two-state model combines moderately elevated and extreme volatility within a broad high-volatility state. The three-state model provides a more granular classification by separating tranquil periods, moderately elevated volatility, and extreme market stress into low-, medium-, and high-volatility states, respectively.

We extract the filtered regime probabilities from the three-state Markov-switching models and incorporate them as regime-dependent features in the subsequent classification analysis (Phase III). To ensure the validity of our out-of-sample predictions, the filtered probabilities are computed using only the training sample and then applied to generate regime probabilities for the test sample. This procedure eliminates look-ahead bias and ensures that the regime-dependent features contain no future information.

Table 6 reports the overall out-of-sample R^2 for the benchmark, two-regime, and three-regime specifications. The benchmark is a regime-agnostic linear HAR-RV-C-CJ-IJ model estimated on the complete training sample and evaluated over the full test sample. The two-regime ensemble estimates separate HAR-RV-C-CJ-IJ models using the low- and high-volatility training observations, whereas the three-regime ensemble estimates separate models using the low-, medium-, and high-volatility training observations. In both ensemble specifications, each test observation is forecast by the model corresponding to its assigned regime, and the resulting forecasts are pooled and evaluated jointly over the full test sample. The two-regime ensemble does not improve point-forecast accuracy on the expanded sample, producing lower out-of-sample R^2 values than the benchmark for all five sectors. In contrast, the three-regime ensemble delivers uniform improvements. The R^2 gain ranges from 0.0616 for XLY (0.5312 to 0.5928) to 0.1555 for XLP (0.4538 to 0.6093). The corresponding overall RMSE reductions range from approximately 6.8% to 15.4% relative to the benchmark. As illustrated by the regime plots, the three-state specification distinguishes intermediate volatility from the relatively small extreme-volatility tail, whereas these observations are combined within a broader state under the two-regime specification. This finer partition appears to allow the regime-specific models to capture state-dependent volatility dynamics more effectively.

Nevertheless, lower forecast errors do not by themselves establish superior portfolio performance, trading profitability, hedging effectiveness, or risk-management performance. Demonstrating such benefits would require a decision-based evaluation incorporating transaction costs, portfolio utility, hedging errors, or the coverage and loss performance of Value-at-Risk or Expected Shortfall forecasts. Moreover, the number of extreme-regime observations in the test sample is relatively small, even after expanding

the sample, so estimated forecasting performance during these periods remains less precise. We therefore interpret the three-regime results as improvements in statistical point-forecast accuracy within the evaluated sample; their economic and risk-management implications remain untested.

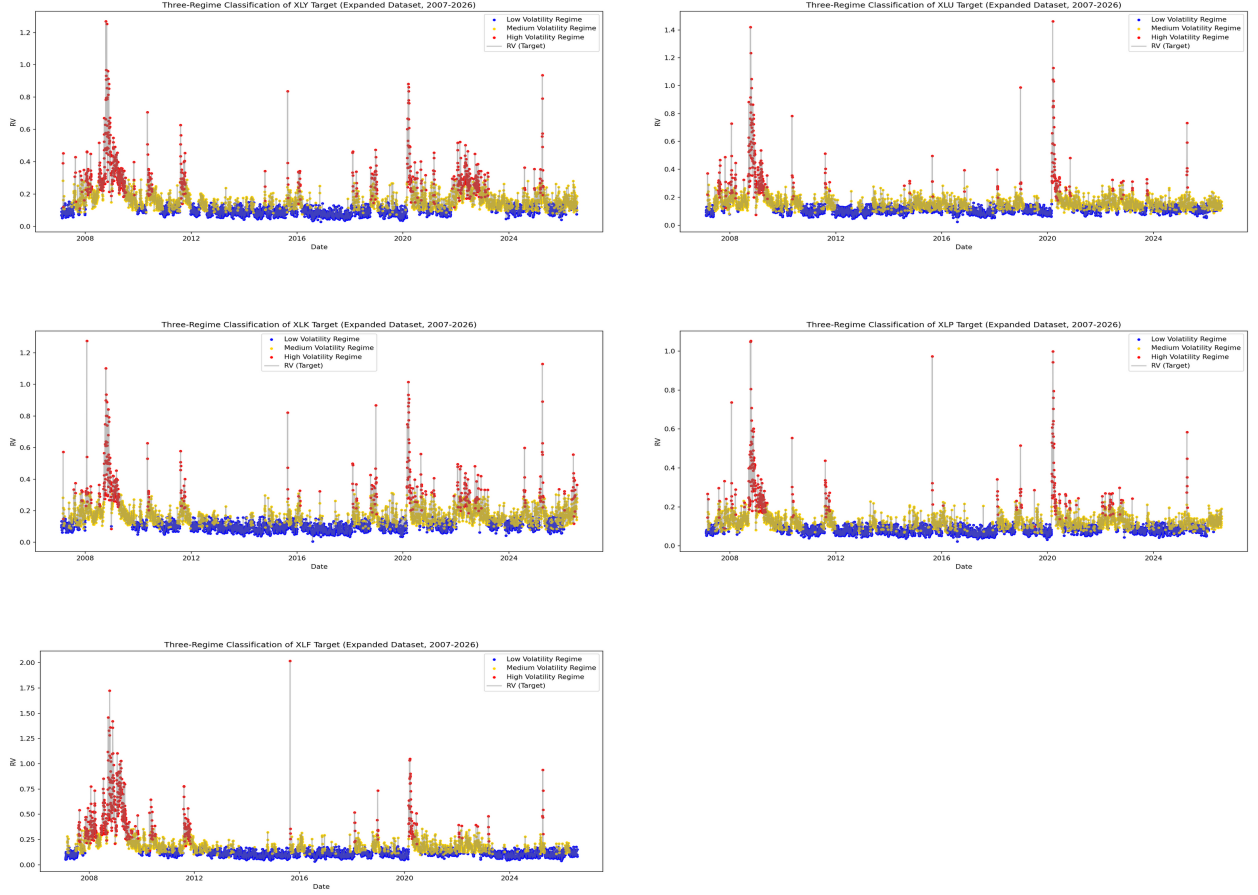


Notes: The figure presents realized volatility (RV) and two-state regime classifications for sector ETFs over the expanded sample period (2007–2026): Consumer Discretionary (XLY), Utilities (XLU), Technology (XLK), Consumer Staples (XLP), and Financials (XLF). Red markers indicate periods classified as high-volatility regimes, while blue markers represent low-volatility regimes. The gray line denotes the realized volatility target series. Regime classifications are obtained from a two-state Markov regime-switching model estimated on RV-based features, capturing volatility clustering and sector-specific risk dynamics over time.

Figure 2: Two-State Volatility Regime Classification.

Table 6: Overall Out-of-Sample R^2 : Benchmark, Two-Regime, and Three-Regime Specifications

Model	XLY	XLU	XLK	XLP	XLF
Benchmark Model	0.5312	0.3777	0.4799	0.4538	0.5148
Two-Regime Ensemble Model	0.5196	0.3377	0.4612	0.4208	0.5124
Three-Regime Ensemble Model	0.5928	0.5285	0.5799	0.6093	0.6250
<i>Notes:</i> The table reports overall out-of-sample R^2 for forecasting realized volatility across five S&P 500 sector ETFs over the expanded 2007–2026 sample.					



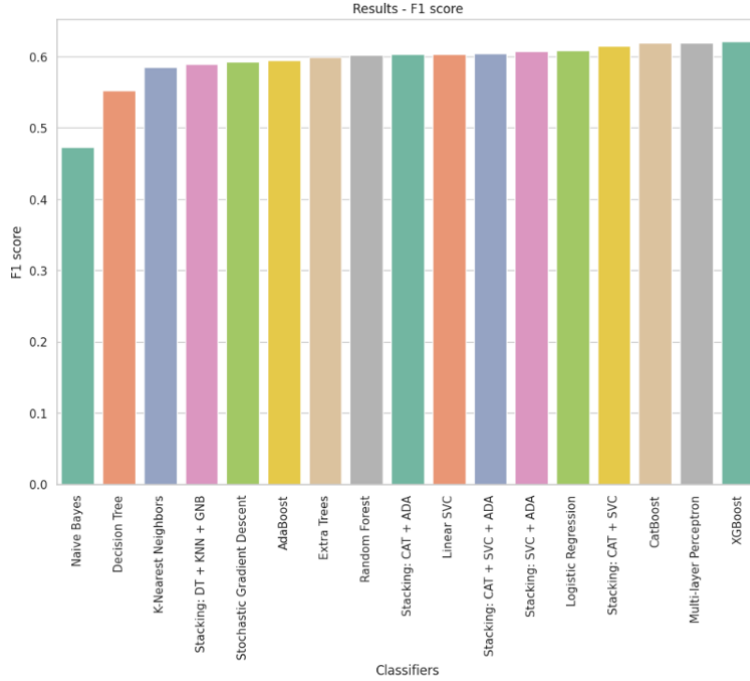
Notes: The figure presents realized volatility (RV) and three-state regime classifications for sector ETFs over the expanded sample period (2007–2026): Consumer Discretionary (XLY), Utilities (XLU), Technology (XLK), Consumer Staples (XLP), and Financials (XLF). Red markers indicate periods classified as high-volatility regimes, yellow markers represent medium-volatility regimes, and blue markers denote low-volatility regimes. The gray line denotes the realized volatility target series. Regime classifications are obtained from a three-state Markov regime-switching model estimated on RV-based features, capturing volatility clustering and sector-specific risk dynamics over time.

Figure 3: Three-State Volatility Regime Classification.

For Phase II, the classification task aims to predict the directional change in realized volatility (RV) from one day to the next. Specifically, the target variable takes the value of 1 if the next day's realized volatility is higher than today's, and 0 otherwise. Formally, the binary response variable Y_{t+1} is defined as:

$$Y_{t+1} = \begin{cases} 1, & \text{if } RV_{t+1} > RV_t, \\ 0, & \text{otherwise.} \end{cases}$$

In this setup, RV_t denotes the realized volatility on day t . Classification accuracy, precision, recall, F1 score, and training run time are reported in Table 7. Figure 4 further illustrates the F1 scores across the models under consideration. It is evident from Table 7 and Figure 4 that XGBoost achieves the highest F1 score (0.621), whereas Naïve Bayes yields the lowest (0.473). XGBoost's superior performance can be attributed to its sequential error-correcting boosting procedure and built-in regularization, which enhance generalization and robustness on this expanded sample.



Notes: Bars report the F1 score (harmonic mean of precision and recall) for binary classification of the direction of realized volatility changes using XLY as the prediction target. Results compare individual classifiers (e.g., Naive Bayes, tree-based models, linear models) and ensemble approaches, including stacking combinations (e.g., CAT, SVC, ADA). Higher values indicate better balance between precision and recall, and the best-performing models are those with the highest bars in the chart.

Figure 4: F1 Score across Classification Models – Prediction Target XLY

For Phase III, we employ the same classification framework to predict the direction of realized volatility using an expanded set of predictors. Specifically, we incorporate regime-sensitive features derived from a three-state Markov regime-switching model, including the estimated probabilities of the high- and medium-volatility regime. These regime-probability features capture nonlinear market dynamics and structural shifts in volatility behavior, enabling the classifiers to adapt more effectively to changing market conditions and potentially improve the accuracy of directional volatility forecasts.

Table 7: Classification Model Performance Comparison – Prediction Target XLY

Model Name	Accuracy	Precision	Recall	F1 Score	Training Time (s)
Logistic regression	0.612	0.628	0.612	0.608	0.071
Stochastic gradient descent	0.609	0.652	0.609	0.592	0.115
Multi-layer perceptron	0.619	0.622	0.619	0.619	5.531
K-nearest neighbors	0.585	0.590	0.585	0.585	0.127
Random forest	0.602	0.603	0.602	0.602	1.518
Extra trees	0.599	0.599	0.599	0.599	0.743
XGBoost	0.621	0.626	0.621	0.621	1.252
Linear SVC	0.608	0.625	0.608	0.603	0.081
Decision tree	0.552	0.553	0.552	0.552	0.163
Naive Bayes	0.527	0.588	0.527	0.473	0.092
AdaBoost	0.594	0.595	0.594	0.595	0.754
CatBoost	0.619	0.623	0.619	0.619	9.285
Stacking: DT + KNN + GNB	0.589	0.594	0.589	0.589	1.528
Stacking: CAT + SVC + ADA	0.605	0.611	0.605	0.604	25.679
Stacking: CAT + SVC	0.615	0.622	0.615	0.614	22.784
Stacking: SVC + ADA	0.608	0.614	0.608	0.608	1.894
Stacking: CAT + ADA	0.603	0.606	0.603	0.603	24.537
<i>Notes:</i> The table reports the out-of-sample performance of classification models used to predict the direction of realized volatility changes for the Consumer Discretionary sector (XLY). Performance metrics include Accuracy, Precision, Recall, and F1 Score. Training time is measured in seconds. Boldface values indicate the best-performing model for each metric. Abbreviations: DT = Decision Tree; KNN = K-Nearest Neighbors; GNB = Gaussian Naive Bayes; CAT = CatBoost; SVC = Support Vector Classifier; ADA = AdaBoost.					

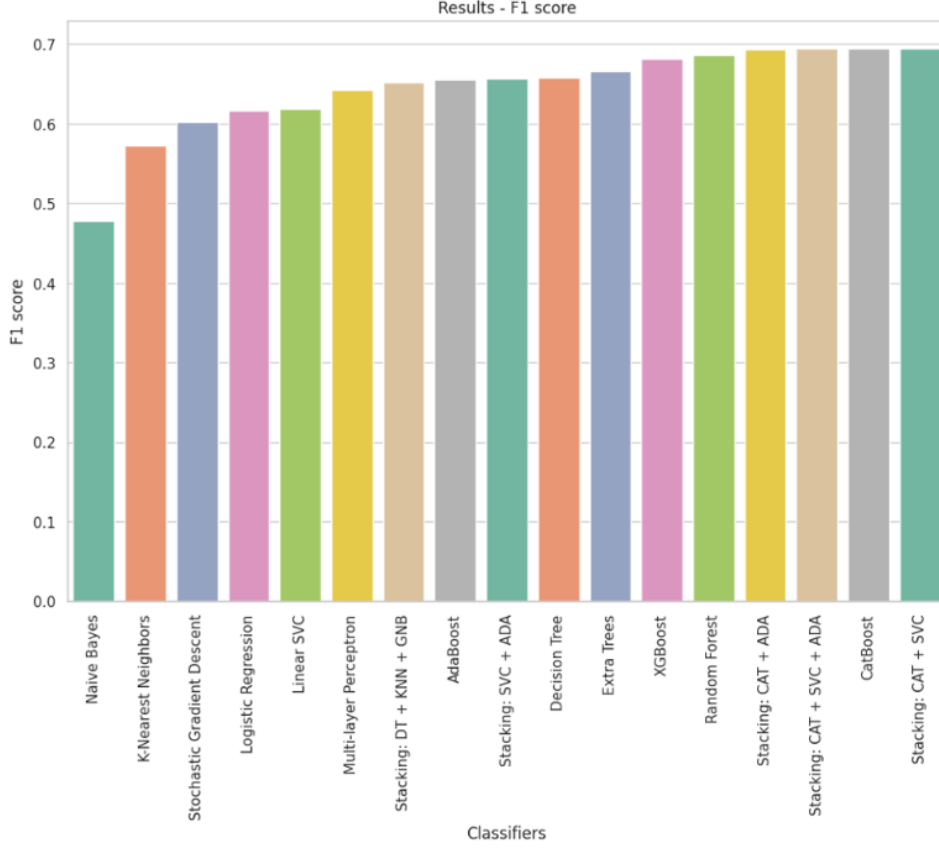
Classification accuracy, precision, recall, F1 score, and training run time are summarized in Table 8, with corresponding F1 scores illustrated in Figure 5. As shown, CatBoost and the two CatBoost-based stacking ensembles (Stacking: CAT + SVC + ADA and Stacking: CAT + SVC) jointly achieve the highest F1 score (0.695), while Naïve Bayes again performs the worst (0.478). The Phase III results show a clear improvement over the Phase II best of 0.621, indicating that the inclusion of regime-sensitive features substantially enhances classification performance. This finding suggests that the probability of high-volatility regimes is a highly informative predictor for forecasting the direction of changes in realized volatility.

Table 8: Classification Model Performance Comparison with Regime Feature – Prediction Target XLY

Model	Accuracy	Precision	Recall	F1 Score	Time Taken (s)
Logistic Regression	0.619	0.632	0.619	0.617	0.210
Stochastic Gradient Descent	0.617	0.658	0.617	0.603	0.104
Multi-layer Perceptron	0.642	0.648	0.642	0.642	7.024
K-Nearest Neighbors	0.573	0.577	0.573	0.573	0.118
Random Forest	0.686	0.687	0.686	0.687	1.552
Extra Trees	0.666	0.666	0.666	0.666	1.000
XGBoost	0.681	0.683	0.681	0.682	3.343
Linear SVC	0.621	0.633	0.621	0.619	0.056
Decision Tree	0.658	0.659	0.658	0.658	0.105
Naïve Bayes	0.530	0.591	0.530	0.478	0.039
AdaBoost	0.655	0.656	0.655	0.655	0.684
CatBoost	0.695	0.698	0.695	0.695	6.598
Stacking: DT + KNN + GNB	0.652	0.652	0.652	0.652	0.462
Stacking: CAT + SVC + ADA	0.695	0.698	0.695	0.695	32.318
Stacking: CAT + SVC	0.695	0.698	0.695	0.695	20.797
Stacking: SVC + ADA	0.657	0.661	0.657	0.657	2.395
Stacking: CAT + ADA	0.694	0.697	0.694	0.694	23.116
<i>Notes:</i> This table reports the out-of-sample performance of classification models incorporating the regime-dependent feature derived from a Markov regime-switching framework. The prediction target is the directional change in realized volatility for the Consumer Discretionary sector (XLY). Performance metrics include Accuracy, Precision, Recall, and F1 Score. Training time is measured in seconds. Boldface values indicate the best-performing model for each metric.					

5.1. Diebold–Mariano Tests of Forecast Accuracy

To address whether the out-of-sample performance differences between the HAR benchmark and the machine-learning regressors are statistically significant, we conduct pairwise Diebold–Mariano (DM) tests [42] comparing the HAR model’s forecast errors against each competing model’s forecast errors over the test sample, for each sector. We report results for the two models that most directly bracket the machine-learning comparison in Table 4: Decision Tree, the weakest-performing individual learner, and Extra Trees, one of the strongest-performing ensemble learners. The null hypothesis is that the two forecasts have equal predictive accuracy; a rejection indicates a statistically significant difference in forecast errors between the two models.



Notes: The figure reports F1 scores for models predicting the direction of next-day realized volatility changes in XLY, using regime-dependent features in addition to the original predictors (Phase III). Results are shown for individual classifiers and ensemble/stacking approaches. Improvements associated with regime information should be assessed relative to the corresponding results without regime features.

Figure 5: F1 Score across Classification Models with Regime Feature – Prediction Target XLY

The Diebold–Mariano results demonstrate that differences in forecast accuracy depend on both the machine-learning method and the sector. In Panel A, HAR has a lower MSE than Decision Tree for all five sectors. The null hypothesis of equal predictive accuracy is rejected at the 5% level for XLY, XLU, XLK, and XLF, but not for XLP ($p = 0.0773$). Thus, the flexibility of an individual decision tree does not translate into improved volatility forecasts relative to the linear HAR benchmark.

Panel B presents a different pattern. Extra Trees produces significantly lower forecast losses than HAR for XLY, XLK, XLP, and XLF. For XLU, however, HAR retains a small but statistically significant forecasting advantage. These results support the sector-dependent findings in the model performance tables: ensemble-based machine-learning methods can generate statistically significant improvements over HAR, but they do not uniformly dominate the linear benchmark across all sectors.

6. Conclusion

This paper examined the forecasting of realized volatility for major sectors of the S&P 500 index by incorporating idiosyncratic and systematic jump risks within a machine learning framework. Motivated by the limitations of traditional econometric models, we evaluated a broad set of ML algorithms and

Table 9: Diebold–Mariano Tests: HAR versus Decision Tree and Extra Trees

Sector	Lower MSE	DM Statistic	p -value	5% Sig.
<i>Panel A: HAR versus Decision Tree</i>				
XLY	HAR	−6.2575	< 0.0001	Yes
XLU	HAR	−2.7727	0.0056	Yes
XLK	HAR	−2.8523	0.0043	Yes
XLP	HAR	−1.7666	0.0773	No
XLF	HAR	−3.2035	0.0014	Yes
<i>Panel B: HAR versus Extra Trees</i>				
XLY	Extra Trees	5.3841	< 0.0001	Yes
XLU	HAR	−2.7010	0.0069	Yes
XLK	Extra Trees	3.8613	0.0001	Yes
XLP	Extra Trees	2.4960	0.0126	Yes
XLF	Extra Trees	3.5208	0.0004	Yes
<i>Notes:</i> This table reports pairwise Diebold–Mariano tests comparing the squared-error losses of the HAR benchmark with Decision Tree (Panel A) and Extra Trees (Panel B), using the common nine-variable RV–CJ–IJ information set. The loss differential is defined as $d_t = e_{\text{HAR},t}^2 - e_{\text{alternative},t}^2$; therefore, a positive statistic favors the alternative model, whereas a negative statistic favors HAR. “5% Sig.” indicates whether the null hypothesis of equal expected predictive loss is rejected at the 5% significance level. The reported p -values are based on two-sided tests.				

extended the analysis by integrating regime-dependent features derived from Markov regime-switching models. The objective was not only to improve volatility forecasts but also to better understand the relative predictive power of different sources of risk.

The empirical findings provide several important insights. First, selected machine-learning models improve forecasting accuracy relative to HAR, but the gains depend on the sector and algorithm, and HAR remains a competitive benchmark. Second, systematic jump measures rank above idiosyncratic jump measures in several specifications, although their relative importance varies across sectors and models. Third, incorporating regime-switching based features significantly improves the prediction of volatility direction. These improvements concern directional forecast accuracy; their implications for portfolio allocation, trading strategies, and risk management require a separate economic evaluation.

Taken together, these results highlight the importance of jointly considering high-frequency jump measures, regime-dependent features, and modern machine learning techniques when modeling volatility. By integrating these elements, we provide a framework that advances both forecasting accuracy and economic interpretability.

7. Limitations and Future Research

This study focuses on one-day-ahead point forecasts of realized volatility and classification forecasts of its directional movement. It therefore does not quantify predictive uncertainty through interval, quantile, or density forecasts. Such forecasts are particularly relevant to risk-management applications because

they describe the range and tail behavior of potential volatility outcomes rather than only their conditional mean or direction. Future research could extend the proposed jump- and regime-based framework using quantile regression, conformal prediction, bootstrap prediction intervals, or probabilistic machine-learning models. These extensions should be evaluated in terms of calibration, coverage, sharpness, and tail performance.

Future work could examine alternative feature-selection procedures, regime-identification thresholds, hyperparameter configurations, and different training–testing periods. Testing the framework across additional asset classes and market conditions would also help determine whether the forecasting gains generalize beyond the sector ETFs and sample considered here.

CRedit authorship contribution statement

First Author: Weijia Peng. Conceptualization, Methodology, Software, Writing – original draft, Writing – review & editing.

Second Author: Eman Abdelfattah. Methodology, Software, Writing – original draft, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

<https://wrds-www.wharton.upenn.edu/pages/about/data-vendors/>.

References

- [1] F. X. Diebold, K. Yilmaz, Measuring financial asset return and volatility spillovers, with application to global equity markets, *The Economic Journal* 119 (534) (2009) 158–171.
- [2] R. F. Engle, Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation, *Econometrica* (1982) 987–1007.
- [3] T. Bollerslev, Generalized autoregressive conditional heteroskedasticity, *Journal of Econometrics* 31 (3) (1986) 307–327.
- [4] F. Corsi, A simple approximate long-memory model of realized volatility, *Journal of Financial Econometrics* 7 (2) (2009) 174–196.
- [5] A. Bucci, Realized volatility forecasting with neural networks, *Journal of Financial Econometrics* 18 (3) (2020) 502–531. doi:10.1093/jjfinc/nbaa008.

- [6] K. Christensen, M. Siggaard, B. Veliyev, A machine learning approach to volatility forecasting, *Journal of Financial Econometrics* 21 (5) (2023) 1680–1727.
- [7] R. Gupta, J. Nel, C. Pierdzioch, Investor confidence and forecastability of us stock market realized volatility: Evidence from machine learning, *Journal of Behavioral Finance* 24 (1) (2023) 111–122.
- [8] W. Peng, C. Yao, Sector-level equity returns predictability with machine learning and market contagion measure: W. peng, c. yao, *Empirical Economics* 65 (4) (2023) 1761–1798.
- [9] C. Zhang, Y. Zhang, M. Cucuringu, Z. Qian, Volatility forecasting with machine learning and intraday commonality, *Journal of Financial Econometrics* 22 (2) (2024) 492–530. doi:10.1093/jjfinec/nbad005.
- [10] T. G. Andersen, T. Bollerslev, F. X. Diebold, Roughing it up: Including jump components in the measurement, modeling, and forecasting of return volatility, *The Review of Economics and Statistics* 89 (4) (2007) 701–720.
- [11] W. Peng, C. Yao, Co-jumps, co-jump tests, and volatility forecasting: Monte carlo and empirical evidence, *Journal of Risk and Financial Management* 15 (8) (2022) 334.
- [12] J. D. Hamilton, A new approach to the economic analysis of nonstationary time series and the business cycle, *Econometrica* 57 (2) (1989) 357–384.
- [13] F. Ma, M. I. M. Wahab, D. Huang, W. Xu, Forecasting the realized volatility of the oil futures market: A regime switching approach, *Energy Economics* 67 (2017) 136–145.
- [14] R. J. Elliott, G.-H. Lian, Pricing variance and volatility swaps in a stochastic volatility model with regime switching: Discrete observations case, *Quantitative Finance* 13 (5) (2013) 687–698.
- [15] X. Lu, F. Ma, J. Wang, J. Liu, Forecasting oil futures realized range-based volatility with jumps, leverage effect, and regime switching: New evidence from midas models, *Journal of Forecasting* 41 (4) (2022) 853–868.
- [16] Y. Shi, Long memory and regime switching in the stochastic volatility modelling, *Annals of Operations Research* 320 (2) (2023) 999–1020.
- [17] J. Luo, T. Klein, Q. Ji, C. Hou, Forecasting realized volatility of agricultural commodity futures with infinite hidden markov har models, *International Journal of Forecasting* 38 (1) (2022) 51–73.
- [18] M. Liu, C.-C. Lee, Capturing the dynamics of the china crude oil futures: Markov switching, co-movement, and volatility forecasting, *Energy Economics* 103 (2021) 105622.
- [19] F. Ma, X. Lu, K. Yang, Y. Zhang, Volatility forecasting: Long memory, regime switching and heteroscedasticity, *Applied Economics* 51 (38) (2019) 4151–4163.
- [20] S. S. Lee, P. A. Mykland, Jumps in financial markets: A new nonparametric test and jump dynamics, *The Review of Financial Studies* 21 (6) (2008) 2535–2563.

- [21] D. Gilder, M. B. Shackleton, S. J. Taylor, Cojumps in stock prices: Empirical evidence, *Journal of Banking and Finance* 40 (2014) 443–459.
- [22] D. B. Nelson, Conditional heteroskedasticity in asset returns: A new approach, *Econometrica* (1991) 347–370.
- [23] L. R. Glosten, R. Jagannathan, D. E. Runkle, On the relation between the expected value and the volatility of the nominal excess return on stocks, *The Journal of Finance* 48 (5) (1993) 1779–1801.
- [24] S. J. Taylor, *Modelling Financial Time Series*, World Scientific, Singapore, 2008.
- [25] T. G. Andersen, T. Bollerslev, Answering the skeptics: Yes, standard volatility models do provide accurate forecasts, *International Economic Review* (1998) 885–905.
- [26] B. Yu, W. Peng, C. Yao, W. Lan, Forecasting realized volatility of shanghai oil futures with mix-frequency uncertainty factors, *Pacific-Basin Finance Journal* 98 (2026) 103150.
- [27] E. Hajizadeh, A. Seifi, M. F. Zarandi, I. Turksen, A hybrid modeling approach for forecasting the volatility of s&p 500 index return, *Expert Systems with Applications* 39 (1) (2012) 431–436.
- [28] Y. Zhang, Y. Peng, Y. Song, Realized volatility forecasting for stocks and futures indices with rolling ceemdan and machine learning models, *Computational Economics* 66 (2) (2025) 1215–1268.
- [29] Z.-C. Li, C. Xie, G.-J. Wang, Y. Zhu, Z.-J. Zeng, J. Gong, Forecasting global stock market volatilities: A shrinkage heterogeneous autoregressive (har) model with a large cross-market predictor set, *International Review of Economics & Finance* 93 (2024) 673–711.
- [30] S.-H. Poon, C. W. J. Granger, Forecasting volatility in financial markets: A review, *Journal of Economic Literature* 41 (2) (2003) 478–539.
- [31] Y. Aït-Sahalia, Disentangling diffusion from jumps, *Journal of Financial Economics* 74 (3) (2004) 487–528.
- [32] X. Huang, G. Tauchen, The relative contribution of jumps to total price variance, *Journal of Financial Econometrics* 3 (4) (2005) 456–499.
- [33] J. Jacod, V. Todorov, Testing for common arrivals of jumps for discretely observed multidimensional processes, *The Annals of Statistics* 37 (4) (2009) 1792–1838.
- [34] T. Bollerslev, V. Todorov, S. Z. Li, Jump tails, extreme dependencies, and the distribution of stock returns, *Journal of Econometrics* 172 (2) (2013) 307–324.
- [35] O. E. Barndorff-Nielsen, N. Shephard, Power and bipower variation with stochastic volatility and jumps, *Journal of Financial Econometrics* 2 (1) (2004) 1–37.
- [36] K. Christensen, R. C. A. Oomen, M. Podolskij, Fact or friction: Jumps at ultra high frequency, *Journal of Financial Economics* 114 (3) (2014) 576–599.

- [37] V. Todorov, G. Tauchen, Volatility jumps, *Journal of Business and Economic Statistics* 29 (3) (2011) 356–371.
- [38] T. Bollerslev, V. Todorov, Tails, fears, and risk premia, *The Journal of Finance* 66 (6) (2011) 2165–2211.
- [39] A. Mukherjee, W. Peng, N. R. Swanson, X. Yang, Financial econometrics and big data: A survey of volatility estimators and tests for the presence of jumps and co-jumps, in: *Handbook of statistics*, Vol. 42, Elsevier, 2020, pp. 3–59.
- [40] R. Gençay, M. Dacorogna, U. A. Muller, O. Pictet, R. Olsen, *An Introduction to High-Frequency Finance*, Elsevier, Amsterdam, 2001.
- [41] C. T. Brownlee, G. M. Gallo, Financial econometric analysis at ultra-high frequency: Data handling concerns, *Computational Statistics and Data Analysis* 51 (4) (2006) 2232–2245.
- [42] F. X. Diebold, R. S. Mariano, Comparing predictive accuracy, *Journal of Business & economic statistics* 20 (1) (2002) 134–144.