Contents lists available at ScienceDirect

International Review of Financial Analysis

journal homepage: www.elsevier.com/locate/irfa

A novel HAR-type realized volatility forecasting model using graph neural network

Nan Hu[a,1], Xuebao Yin[a,1,*], Yuhang Yao[b]

[a] *School of Economics and Finance, South China University of Technology, Guangzhou 510006, China*
[b] *Lingnan College, Sun Yat-sen University, Guangzhou 510275, China*

ARTICLE INFO

JEL classification:
C18
C53
C45
C58

Keywords:
Volatility forecasting
Graph neural network
Heterogeneous autoregression
Machine learning
Deep learning

ABSTRACT

This study introduces a novel model that uses the convolutional neural network (CNN) architecture to fully utilize information from the heterogeneous autoregressive (HAR) family components across different windows in a two-dimensional image format, to forecast the direction of stock market volatility. The proposed model, Convolutional Neural Network-based Heterogeneous Autoregressive-Kitchen Sink (CNN-HAR-KS) model, leverages the CNN model's automated signal generation capabilities while incorporating the HAR components to ensure interpretability and comparability with traditional volatility forecasting models. Empirical analysis in China's stock market reveals that in terms of forecast accuracy, the CNN-HAR-KS model outperforms alternative models, such as logistic or machine learning models, using the same input variables. We construct investment portfolios based on realized volatility forecasts to further explore the CNN-HAR-KS model's economic value. The CNN-HAR-KS model yields the highest daily Sharpe ratio of 0.043, indicating its ability to generate better risk-adjusted returns than alternative models. Further analysis suggests that the proposed model can be applied to other realized volatility-related classification problems.

1. Introduction

Accurate volatility modeling and forecasting are critical for asset allocation, risk management, and option pricing. GARCH-type models, first proposed by Engle (1982) and Bollerslev (1986), have long been used to forecast stock market volatility (Fang et al., 2020; Wang et al., 2020; Yang et al., 2020). However, these models are based on daily returns and do not consider high-frequency data. Furthermore, they view volatility as a latent process, making it difficult to model directly (Chen et al., 2018; Liu et al., 2018; Wilms et al., 2021).

Recent advancements in information technology have made it possible to exploit intraday information from high-frequency data. Consequently, high-frequency realized volatility (RV), defined as the sum of squared intraday high-frequency returns, can be directly modeled using available intraday information (Andersen et al., 2001, 2005, 2007, 2012; Andersen & Bollerslev, 1998; Barndorff-Nielsen & Shephard, 2004; Corsi, 2009; Koopman et al., 2005; Patton & Sheppard, 2015), rather than being treated as a latent process in GARCH-type models (Wilms et al., 2021; Wang et al., 2020). Notably, numerous methods have been proposed to forecast realized volatility (Chen et al., 2018; Izzeldin et al., 2019; Luo et al., 2022; Motegi et al., 2020; Xiao et al., 2021).

Among the RV models, the heterogeneous autoregressive realized volatility (HAR-RV) model, introduced by Corsi (2009), has gained widespread recognition for its efficacy in volatility forecasting. The HAR-RV model shows tractable estimation, high parsimony, and the ability to capture long memory in volatility sequences (Wang et al., 2016). Consequently, it has emerged as the standard benchmark for volatility prediction in academic literature (Gong & Lin, 2019; Izzeldin et al., 2021; Liu et al., 2018; Luo et al., 2022; Wen et al., 2019).

Building upon the HAR-RV model, several studies have proposed meaningful extensions. For example, Andersen et al. (2007) introduced the jump component to the HAR model, resulting in the HAR-RV-Jump (HARJ) model. Furthermore, Andersen et al. (2012) incorporated the realized tri-power quarticity to model and forecast volatility. Patton and Sheppard (2015) also decomposed RV into positive and negative semi-variance based on intraday return signs. Within specific sample contexts, integrating these distinct components resulted in improved volatility forecasting accuracy when compared to the benchmark HAR-RV model.

[*] Corresponding author.
E-mail address: 202310190525@mail.scut.edu.cn (X. Yin).
[1] First author and second author contribute equally to this work.

https://doi.org/10.1016/j.irfa.2024.103881
Received 12 March 2024; Received in revised form 15 December 2024; Accepted 15 December 2024
Available online 20 December 2024
1057-5219/© 2024 Published by Elsevier Inc.

However, how to select the most suitable model based on market conditions remains unclear. The challenge is to effectively leverage the abundant information embedded in the various HAR-type components (i.e., jump, Jo jump, and BNS jump) while addressing the issue of multicollinearity. Furthermore, linear extensions of the HAR-type model do not account for the complex and nonlinear interrelationships among diverse variables.

Machine learning (ML) introduces a novel methodological framework for uncovering complex relationships among variables, demonstrating superior predictive performance for asset returns and volatility relative to conventional modeling approaches. (Audrino & Knaus, 2016; Audrino et al., 2020; Carr et al., 2019; Erel et al., 2021; Gu et al., 2020; Leippold et al., 2022; Luo et al., 2024; Song et al., 2024; Su et al., 2023; Vrontos et al., 2021). Considering the lack of interpretability in ML's black-box mechanisms, various scholars seek to combine ML with HAR-type components, specifically lagged RV, to ensure model interpretability and comparability with the traditional HAR model. For example, Rahimikia and Poon (2020) compare ML and HAR models for forecasting RV, using variables from limit-order books and news. Similarly, studies show that external information from investor site visits can enhance forecast quality, highlighting the potential of integrating diverse data sources into predictive models (Guo et al., 2024). Moreover, Christensen et al. (2023) propose an ML-based model that uses HAR-type components, firm-specific characteristics, and macroeconomic indicators to forecast one-day-ahead RV. Meanwhile, Zhang et al. (2023) leverage ML techniques to predict intraday volatility dynamics and provide empirical evidence that their proposed model has superior out-of-sample predictive accuracy.

Furthermore, a substantial body of research indicates that deep learning (DL) architectures significantly improve out-of-sample predictive accuracy compared to other ML methods (Audrino & Knaus, 2016; Bucci, 2020; Chen et al., 2024). The ability of DL models to flexibly approximate highly complex functional forms contributes to their superior performance. These studies primarily focus on the family, which can effectively capture the long-memory properties inherent in volatility sequences. Convolutional neural networks (CNNs) are primarily used for image and video processing, recognition, and classification, so their application in volatility forecasting is limited. Jiang et al. (2023) employ a CNN to model the predictive relationship between images and future stock return directions. Their proposed CNN-based model exhibits superior accuracy in return predictions when compared to conventional methods.

The RV classification problems are equally important and have been explored in previous studies (BenSaida, 2021; Bollerslev et al., 2020; Patton & Sheppard, 2015; Zhang & Zhao, 2023). From a theoretical perspective, RV classification and prediction based on historical returns can be viewed as an extension of return extrapolation, a new and cutting-edge branch of behavioral finance (Atmaz, 2022; Li & Liu, 2023; Lochstoer & Muir, 2022; Xie et al., 2023). Specifically, Atmaz's (2022) modeling mechanism provides an intuitive motivation, suggesting that positive (negative) stock returns would reinforce investors' beliefs regarding higher volatility in future returns. This insight motivates us to predict the direction of volatility and serves as the foundation for developing investment strategies (Chen & Petkova, 2012; Gu et al., 2018; Guo & Savickas, 2010). Moreover, we apply this forecasting framework to other scenarios, such as predicting good/bad volatility and negative jumps.

To improve the accuracy of stock market volatility direction forecasts, this study introduces a novel approach that combines CNNs with the HAR model family, represented in a two-dimensional (2D) image format. Our proposed model relies on the advantageous characteristics of CNNs in image classification and analysis tasks, which are applied to various HAR-type components to generate feature matrices in the form of two-dimensional images. We improve our ability to predict RV by combining CNNs' superior image classification performance with HAR-type components.

We contribute to the literature in the following aspects: First, we propose a novel volatility forecasting model, namely, the CNN-HAR-KS model, which takes advantage of CNN models' image learning capabilities to investigate the predictive power of HAR-type components for RV in 2D image format. We specifically construct 16 HAR-type components (i.e., RV, BPV, ABD jump, ABD CSP, BNS jump, BNS CSP, Jo jump, Jo CSP, RS+, RS−, Daily return, Negative RV, SJ, SJ+, SJ−, and TQ), including their one-day lag and moving averages ranging from 6 to 20 days. Subsequently, we transform the dataset into a 16 × 16 matrix with daily, weekly, and monthly data for the HAR-type components. Finally, we convert the matrix to images and use a CNN to extract the complex relationships between the HAR-type components and the target variable. This framework is called the CNN-HAR-KS model because it includes all HAR-type components, resembling a so-called "kitchen sink model."

Second, we compare forecasting performance of HAR-type logistic, ML-based, and DL-based HAR-type models. Our empirical findings unequivocally demonstrate the CNN-HAR-KS model's superior forecast accuracy. Notably, this model uses image-formatted HAR-type components as inputs, thereby showcasing its enhanced predictive capabilities.

Third, the proposed CNN-HAR-KS model has the potential to generate substantial economic value. Leveraging the low volatility anomaly observed in asset pricing literature (Chen & Petkova, 2012; Gu et al., 2018; Guo & Savickas, 2010), we propose an investment strategy that entails holding long and short positions in asset portfolios with lower and higher volatilities, respectively. The primary goal of this strategy is to exploit the low volatility anomaly and achieve superior risk-adjusted returns. Our findings show that the CNN-HAR-KS model achieves the highest daily Sharpe ratio of 0.043, resulting in an impressive cumulative return of 185.522 % over the entire test dataset (1000 trading days). Furthermore, the model exhibits significantly lower drawdowns than its competing counterparts.

The remainder of this study proceeds as follows. Section 2 provides a detailed description of the study's methodology. Section 3 describes the dataset. Section 4 presents the empirical findings, including initial forecasts and the robustness test results. Section 5 explores the economic value of the proposed models. Section 6 investigates the proposed model's potential application to alternative classification problems. Finally, Section 7 concludes the study.

2. Methodology

2.1. Realized volatility direction measure

In their seminal contribution, Andersen and Bollerslev (1998) proposed a groundbreaking approach that advocates using RV as a suitable proxy for integrated variance. To calculate the RV for a given trading day t, we add the squared intraday returns, denoted as $r_{t,j}$. The RV can be calculated as follows:

$$RV_t = \sum_{j=1}^{M} r_{t,j}^2 \qquad t = 1, 2, 3, \ldots T, \tag{1}$$

where j represents the intraday time intervals, $1/M$is the given sampling frequency.

Initially, we compute the daily RV using the 5-min closing prices. Subsequently, because our goal is to forecast the direction of volatility to construct a portfolio based on low volatility anomalies, we define the RV direction as follows:

$$RVD = \begin{cases} 1, & if \quad \widehat{RV}_t - RV_{t-1} < 0 \\ 0, & if \quad \widehat{RV}_t - RV_{t-1} \geq 0 \end{cases} \tag{2}$$

where the $\widehat{RV}_t$ denotes the forecasts generated by the models at time t and RV_{t-1} means the true RV at time $t-1$.

2.2. Logistic HAR-type models

We use logistic regression as the benchmark model. We construct several logistic-HAR-type models by selecting a diverse range of logistic-HAR components to investigate the impact of different HAR components on volatility forecasting. Initially, we construct the standard HAR specification using a similar approach to Corsi (2009).

$$RVD_{t+1} = logit(p) = \beta_0 + \beta_d RV_t + \beta_w RV_{w,t} + \beta_m RV_{m,t} + \epsilon_{t+1}, \tag{3}$$

where $RV_{w,t}$ and $RV_{m,t}$ denote the average RV from day $t-4$ to day t and from day $t-21$ to t, respectively.

Andersen et al. (2007) extend the standard HAR model to the HARJ model, which incorporates the jump component. The logistic-HARJ specifications are as follows:

$$RVD_{t+1} = logit(p) = \beta_0 + \beta_d RV_t + \beta_w RV_{w,t} + \beta_m RV_{m,t} + \beta_{j,d} J_t + \epsilon_{t+1}, \tag{4}$$

where J_t is the jump component, namely $J_t = max(RV_t - BVP_t, 0)$, $BPV_t = u_1^{-2} \sum_{j=2}^{M} |r_{t,j-1}||r_{t,j}|$ and $u_1 = (2/\pi)^{0.5} = E(|Z|)$.

We know that the RV of the logistic-HARJ model has two components: continuous and jump. This RV decomposition extends to the logistic-HARCJ model.

$$\begin{aligned} RVD_{t+1} = logit(p) = \beta_0 + \beta_{c,d} CSP_t + \beta_{c,w} CSP_{w,t} + \beta_{c,m} CSP_{m,t} + \\ + \beta_{j,d} CJ_t + \beta_{j,w} CJ_{w,t} + \beta_{j,m} CJ_{m,t} + \epsilon_{t+1}, \end{aligned} \tag{5}$$

where $CSP_t = I(Z_t \le \Phi_\alpha) \cdot RV_t + I(Z_t > \Phi_\alpha)$, the$I(\cdot)$ is the indicator function.$CJ_t = I(Z_t > \Phi_\alpha) \cdot [RV_t - BPV_t]$. The $CSP_{w,t}(CJ_{w,t})$ and $CSP_{m,t}(CJ_{m,t})$are the weekly and monthly averages of the continuous component (the jump component), respectively.

Patton and Sheppard (2015) present a novel HAR-RV model based on signed jump variation. This new model divides the RV into two semivariances (HARRSI) to account for asymmetric volatility. The logistic-HARRSI model has the following specifications:

$$RVD_{t+1} = logit(p) = \beta_0 + \beta_d^+ RS_t^+ + \beta_d^- RS_t^- + \beta_w RV_{w,t} + \beta_m RV_{m,t} + \epsilon_{t+1}, \tag{6}$$

where $RS_t^- = \sum_{j=1}^{M} r_{t,j}^2 I(r_{t,j} < 0)$ and $RS_t^+ = \sum_{j=1}^{M} r_{t,j}^2 I(r_{t,j} \ge 0)$.

The logistic-HARRSI model can be extended to the logistic-HARRSII model by including the interaction between lagged RV and negative returns.

$$\begin{aligned} RVD_{t+1} = logit(p) = \beta_0 + \beta_d^+ RS_t^+ + \beta_d^- RS_t^- + \gamma RV_{d,t} I(r_t < 0) \\ + \beta_w RV_{w,t} + \beta_m RV_{m,t} + \epsilon_{t+1}, \end{aligned} \tag{7}$$

To capture the leverage effect including a signed jump variation and a bi-power variation, Patton and Sheppard (2015) proposed the HARRSJI model to capture both the leverage effect and a bi-power variation.

$$RVD_{t+1} = logit(p) = \beta_0 + \beta_{j,d} SJ_t + \beta_{bv,d} BV_t + \beta_w RV_{w,t} + \beta_m RV_{m,t} + \epsilon_{t+1}, \tag{8}$$

where $SJ_t = RS_t^+ - RS_t^-$.

The logistic-HARRSJII model is an extension of the logistic-HARRSJI model; it can capture the leverage effect and distinguish between the roles of positive and negative jumps.

$$\begin{aligned} RVD_{t+1} = logit(p) = \beta_0 + \beta_{j,d}^+ SJ_t^+ + \beta_{j,d}^- SJ_t^- + \beta_{bv,d} BV_t \\ + \beta_w RV_{w,t} + \beta_m RV_{m,t} + \epsilon_{t+1}, \end{aligned} \tag{9}$$

where $SJ_t^+ = SJ_t \cdot I(SJ_t > 0)$ and $SJ_t^- = SJ_t \cdot I(SJ_t < 0)$.

Finally, we construct a “kitchen sink model,” also known as the logistic-HAR-KS model, using all of the HAR-type components, including RV, BPV, ABD jump, and ABD continuous components. In particular, we account for heterogeneous beliefs by incorporating a one-day lag and moving averages of HAR-type components ranging from 6 to 20 days.

Table 1
HAR-type components.

Variable name	Description	Reference
RV	Realized volatility	Andersen and Bollerslev (1998)
BPV	Realized bi-power variance	Barndorff-Nielsen and Shephard (2004)
ABD jump	Jump component using the ABD jump test	Andersen et al. (2007)
ABD CSP	Continuous component using ABD jump test	Andersen et al. (2007)
BNS jump	Jump component using the BNS jump test	Barndorff-Nielsen and Shephard (2006)
BNS CSP	Continuous component using BNS jump test	Barndorff-Nielsen and Shephard (2006)
Jo jump	Jump component using the Jo jump test	Jiang and Oomen (2008)
Jo CSP	Continuous component using Jo jump test	Jiang and Oomen (2008)
RS+	RV using positive 5-min returns	Patton and Sheppard (2015)
RS−	RV using negative 5-min returns	Patton and Sheppard (2015)
Daily return	Daily return	
Negative RV	RV times an indicator for the negative daily returns	Patton and Sheppard (2015)
SJ	RS+ minus RS−	Patton and Sheppard (2015)
SJ+	SJ times an indicator for the positive SJ	Patton and Sheppard (2015)
SJ−	SJ times an indicator for the negative SJ	Patton and Sheppard (2015)
TQ	Realized tri-power quarticity	Andersen et al. (2012)

$$RVD_{t+1} = logit(p) = \beta_0 + \beta_d \mathbf{X_{t-1}} + \sum_{i=6}^{20} \beta_i \mathbf{X_{t-i}} + \epsilon_{t+1}, \tag{10}$$

where the $\mathbf{X_t}$ denotes the HAR-type components vector. We detail these measures in Table 1.

2.3. Machine learning-based HAR-type model

The logistic models represented by Eqs. (3)–(10) are based on the underlying assumption that the HAR-type components are linearly separable. However, when there is a high level of correlation or nonlinear relationships between features, the logistic regression model may struggle to accurately capture these complex dynamics, resulting in poor predictive performance. Therefore, we employ the random forest, LightGBM, and XGBoost algorithms, respectively, and the HAR-type components are used as input variables to construct ML-based HAR-type models. By integrating these ML techniques, we aim to enhance the models' predictive capabilities and forecast accuracy.

2.3.1. RF-HAR-KS model

The random forest (RF) model, first proposed by Breiman (2001), is an improved version of the decision tree algorithm. It finds extensive use in addressing classification and regression problems and necessitates the optimization of several parameters. Within the RF model, two parameters significantly affect performance, namely, the number of trees (n_{tree}) and the number of randomly sampled candidate variables at each split (n_{try}). The value of n_{try} is often suggested to be$p/3$, where$p$ denotes the number of input variables (Dudek, 2015). In forecasting the direction of RV (RVD_t) at time t, we employ the RF model with all HAR-type components as inputs, denoted as RF-HAR-KS.

2.3.2. LightGBM-HAR-KS model

We further consider the Light Gradient Boosting Machine (LightGBM) algorithm for extracting information from the HAR-type component. The LightGBM algorithm is a novel gradient boosting decision tree (GBDT) algorithm proposed by Ke et al. (2017) that has been used in a wide range of data mining tasks, including classification, regression, and ordering. The LightGBM algorithm implements two novel techniques, namely, gradient-based one-sided sampling and exclusive feature bundling. Appendix A contains a detailed specification of the model. With all of the HAR-type components as input, we refer to the LightGBM-HAR-KS model.

2.3.3. XGBoost-HAR-KS model

Xtreme gradient boosting (XGBoost), a form of gradient boosting (GB) with regularization, provides an improved regularization framework that effectively reduces the risk of overfitting (Chen & Guestrin, 2016). The methodology is derived from the fundamental concept of GB, in which an ensemble model is constructed by combining K additive trees. This method ensures that the model construction process is more robust and reliable. Appendix B contains a detailed specification of the model. The model is known as the XGBoost-HAR-KS model because it accepts all HAR-type components as input.

2.4. Deep learning-based HAR-type models

2.4.1. Feed-forward networks

To comprehensively explore the realm of ML, we use artificial neural networks. Neural networks have proven to be extremely adaptable and nonlinear in their ability to solve complex problems. We first use the feed-forward network (FFN) to forecast the direction of the stock's volatility, taking all HAR-type components as input.

A neural network consists of three layers: input, hidden, and output. Let L represent the total number of layers in the network. In the initial layer, denoted as the first layer, the neural network receives an input $\mathbf{X_t}$. The data then transforms one or more hidden layers using an activation function g. The mathematical representation for the l-th layer is as follows:

$$a_t^{\theta_{l+1}b_{l+1}} = g_l\left(\sum_{j=1}^{N_t}\theta_j^{(l)a_t^{\theta_l,b_l}} + b^{(l)}\right), \qquad 1 \le l \le L, \tag{11}$$

where g_l is the activation function, θ^l is the weight matrix, b^l is the error serving as an activation threshold for the neurons in the next layer, N_t denotes the number of hidden neurons, and $a_t^{\theta_{l+1}b_{l+1}}$ is the prediction.

The output layer constructs a forecast as follows:

$$\widehat{f}(\mathbf{X_t}) = (g_L o \dots o g_1)(\mathbf{X_t}), \tag{12}$$

The neural network generates forecasts via a series of functional transformations. According to Cybenko's (1989) universal approximation theorem, a single hidden layer with a sufficiently large number of neurons (and an appropriate activation function) can approximate any continuous function. However, it is frequently more convenient and computationally efficient to include additional hidden layers instead of arbitrarily increasing the number of neurons within a single layer. Consequently, the optimal architecture of a neural network is determined by a tuning process specific to the problem at hand.

We construct the FFN model using a geometric pyramid-inspired architecture (Gu et al., 2020; Masters, 1993). The FFN model comprises five layers with 128, 64, 64, 32, and 16 neurons. Regularization is critical for addressing the problem of overfitting in neural networks. Extensive research has been conducted on various regularization techniques to determine the optimal weights. This study combined two commonly used regularization techniques, namely learning rate shrinkage and early stopping. Appendix C presents the selection of parameters is presented in detail. We determined the optimal parameters by evaluating the validation set loss.

2.4.2. Graph convolutional neural networks (CNN) with HAR-type components

We employ the CNNs to construct our primary model CNN-HAR-KS. First, for each day, RV, BPV, ABD jump, ABD CSP, BNS jump, BNS CSP, Jo jump, Jo CSP, RS+, RS−, Daily return, Negative RV, SJ, SJ+, SJ−, and TQ values are computed for various intervals (1 and 6–20 days). Subsequently, for each day, a 16 × 16 image is created by combining 16 HAR-type components with 16 different intervals of components (Table 2). Meanwhile, each image employs the associated label from Eq. (2). Finally, the CNN technique is used to extract the HAR-type component image. Our proposed CNN analysis phase includes seven layers, namely, an input layer (16 × 16), two convolutional layers (16 × 16 × 32 and 16 × 16 × 64), a max-pooling (8 × 8 × 64), one dropout (0.3), a fully connected layer (64), and an output layer (2). We use the 3 × 3 filter size for the CNN filter due to its convolution capability with the closest neighbors (upper, lower, right, left, upper left, upper right, lower left, and lower right) while processing the current layer.

Table 2
Model specifications.

Model name	Description
logistic-HAR	The logistic regression model that includes the daily, weekly, and monthly average of RVs
logistic-HARJ	The logistic model including daily, weekly, monthly average of RVs, and daily jump
logistic-HARCJ	The logistic model including daily, weekly, monthly average of the continuous sample path and jump
logistic-HARRSI	The logistic model including daily RS+/RS−, weekly, and monthly average of RVs
logistic-HARRSII	The logistic-HARRSI model adds a term that interacts with the lagged RV with an indicator for the negative daily returns
logistic-HARSJI	The logistic regression model that includes daily signed jump and bi-power variation, weekly, and monthly RVs
logistic-HARSJII	The logistic-HARRSI model decomposes the signed jump variation into positive and negative jumps
logistic-HAR-KS	The logistic regression model that includes all HAR components
RF-HAR-KS	Random forest model that includes all HAR-type components
LightGBM-HAR-KS	LightGBM model that includes all HAR components
XGBoost-HAR-KS	XGBoost model that includes all HAR components
FFN-HAR-KS	FFN model that includes all HAR-type components
CNN-HAR-KS	CNN model that includes all HAR components

Our proposed CNN structure includes four layers: convolutional, max-pooling, dropout, and a fully connected multi-layer perceptron layer. The convolutional layer consists of convolution operations. In statistical terms, a basic convolution operation can be defined as follows:

$$s(t) = (x \cdot w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a), \tag{13}$$

The convolution operation is implemented on two-dimensional images as follows:

$$s(i,j) = (I \cdot K)(i,j) = \sum_m \sum_n I(m,n)K(i-m,j-n), \tag{14}$$

where I denotes the input image and K denotes the kernel. Besides, consecutive convolutional and max-pooling layers build the deep neural network structure as follows:

$$e_i = \sum_j W_{i,j}\mathbf{X}_j + b_i, \tag{15}$$

where W denotes weights, $\mathbf{X}$ denotes the input of the HAR-type components, and b denotes bias. At the end of the network, the *softmax* function is used to get the volatility direction:

$$RVD = softmax(e), \tag{16}$$

We find the best parameters by minimizing the loss in the validation set. Following Jiang et al. (2023), we used the standard objective function for classification problems, a cross-entropy loss function, as our evaluation metric. Appendix C provides a detailed presentation of the

Table 3
Descriptive statistics of HAR-type components.

	RV	BPV	Jump	CSP	RS+	RS−	SJ	SJ+	SJ−
Mean	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Std	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Min	0.00	0.00	0.00	0.00	0.00	0.00	−0.01	0.00	−0.01
Max	0.01	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00
Skew	10.02	8.59	17.76	8.58	14.07	14.57	−14.19	22.3	−20.98
Kurt	135.43	97.63	399.95	97.85	315.89	296.72	382.57	729.36	539.85
ACF(1)	0.55***	0.68***	0.16***	0.69***	0.49***	0.35***	0.01***	0.01***	0.10***
Q(5)	2833.71***	4065.59***	250.22***	4689.79***	2701.01***	919.79***	92.11***	261.41***	5.22***
Q(21)	5124.25***	9551.32***	380.85***	9724.08***	4624.97***	1821.62***	106.45***	290.21***	100.58***

Note: Std is standard deviation, skew denotes skewness, Kurt is Kurtosis, Q(5) and Q(21) denote the 5th and 21st order Ljung–Box Q statistics, respectively. *** star represents the rejection of the null hypothesis at a significance level of 1 %.

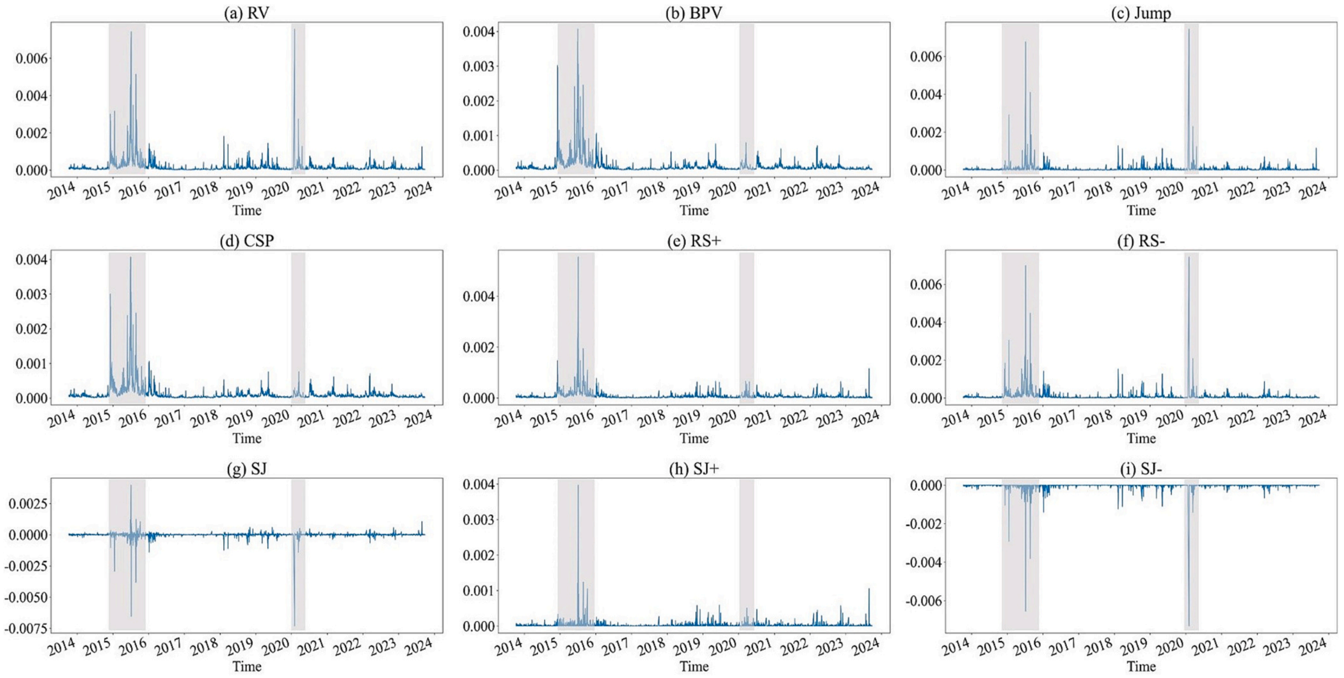

Fig. 1. Evolution of heterogeneous autoregressive (HAR)-type components.

parameter set.

2.5. Performance evaluation

In this paper, we evaluate and compare the performance of stock volatility direction prediction models based on accuracy and area under the receiver operating characteristic (ROC) curve. Accuracy is the proportion of correctly predicted directions. The ROC curve depicts the relationship between the true positive rate (the proportion of correctly predicted volatility directions) and the false positive rate (the proportion of incorrectly predicted volatility directions) at different probability thresholds. The area under the ROC curve (AUC) is a threshold-insensitive metric that measures the model's ability to distinguish between categories (Fawcett, 2006). A higher AUC score indicates that the model is better at distinguishing between volatility directions.

3. Data

We collect 5-min data on the China Securities Index (CSI) 300 and closing prices of individual stocks from November 1, 2013, to September 28, 2023, a total of 2418 trading days. Furthermore, we use the HAR model, as described in Table 3, to calculate daily RV and other components for the CSI 300 index and its constituent stocks. We choose the CSI 300 index constituent stocks to construct our investment portfolio, excluding those added or removed during the observation period. We keep 155 stocks for further analysis and portfolio construction. The dataset has been divided into two subsets, namely, an initial training subsample that covers 1418 days from 2013.11.1 to 08/19/2019 and a complete testing subsample that spans 1000 days from 08/20/2019 to 09/28/2023.[2] To implement the sliding window with retraining methodology, we initially chose training and testing periods of 1418 and 250 days, respectively, from 11/01/2013 to 08/19/2019 for training and 08/20/2019 to 08/28/2020 for testing. Thus, the training and testing periods are moved forward by 250 days. This retrained the model, which was then tested over 250 days, that is, the training period became 11/14/2014 to 08/20/2020 with the testing period set as 08/21/2020 to 09/072020. Thus, each 250-day from 08/20/2019 to 09/28/2023 is used as testing samples. For brevity, Table 3 only provides descriptive statistics for HAR-type components based on the CSI 300 index's 5-min high-frequency closing price. All of the indicators are leptokurtic and fat tail, with a non-normal distribution. Moreover, the results show

[2] The main result shows a roughly 60:40 split between the training and testing samples. To ensure that the results are robust, we use the last 750 days (approximately 70:30) and the last 500 days (approximately 80:20) as prediction test sets.

Table 4
Out-of-sample forecasting for CSI300 index volatility.

	Accuracy					AUC				
	Total Sample	Subsample 1	Subsample 2	Subsample 3	Subsample 4	Total Sample	Subsample 1	Subsample 2	Subsample 3	Subsample 4
Panel A: HAR family logistic forecasting model										
Logistic-HAR	0.63	0.60	0.61	0.68	0.64	0.64	0.59	0.62	0.69	0.66
Logistic-HARJ	0.64	0.60	0.66	0.66	0.63	0.64	0.59	0.66	0.66	0.65
Logistic-HARCJ	0.63	0.58	0.62	0.68	0.66	0.64	0.58	0.63	0.68	0.67
Logistic-HARRSI	0.63	0.59	0.58	0.66	0.67	0.63	0.59	0.59	0.67	0.68
Logistic-HARRSII	0.64	0.61	0.67	0.67	0.60	0.64	0.61	0.67	0.67	0.62
Logistic-HARSJI	0.59	0.54	0.57	0.66	0.58	0.59	0.53	0.58	0.66	0.59
Logistic-HARSJII	0.63	0.59	0.63	0.67	0.62	0.63	0.58	0.64	0.68	0.63
Logistic-HAR-KS	0.65	0.62	0.66	0.70	0.61	0.65	0.61	0.66	0.70	0.62
Panel B: Machine learning forecasting model										
RF-HAR-KS	0.74	0.73	0.75	0.74	0.75	0.74	0.73	0.75	0.73	0.75
LightGBM-HAR-KS	0.75	0.74	0.76	0.78	0.72	0.75	0.74	0.77	0.78	0.72
XGBoost-HAR-KS	0.74	0.72	0.72	0.78	0.74	0.74	0.72	0.73	0.78	0.74
Panel C: Deep learning forecasting model										
FFN-HAR-KS	0.75	0.74	0.76	0.76	0.75	0.75	0.74	0.76	0.76	0.75
CNN-HAR-KS	**0.78**	**0.78**	**0.78**	**0.80**	**0.76**	**0.78**	**0.78**	**0.78**	**0.80**	**0.76**

Note: The bold numbers in the table indicate that the model yields the highest forecasting accuracy.

Table 5
Out-of-sample forecasting performance at individual stock.

	Accuracy					AUC				
	Total Sample	Subsample 1	Subsample 2	Subsample 3	Subsample 4	Total Sample	Subsample 1	Subsample 2	Subsample 3	Subsample 4
Panel A: HAR family logistic forecasting model·										
Logistic-HAR	0.60	0.61	0.62	0.60	0.58	0.60	0.60	0.61	0.59	0.57
Logistic-HARJ	0.61	0.62	0.63	0.61	0.59	0.61	0.61	0.62	0.60	0.58
Logistic-HARCJ	0.61	0.61	0.63	0.61	0.59	0.60	0.61	0.62	0.61	0.58
Logistic-HARRSI	0.61	0.61	0.63	0.61	0.59	0.60	0.61	0.62	0.60	0.58
Logistic-HARRSII	0.61	0.61	0.63	0.61	0.59	0.60	0.61	0.62	0.61	0.58
Logistic-HARSJI	0.57	0.57	0.58	0.57	0.55	0.56	0.56	0.58	0.56	0.55
Logistic-HARSJII	0.61	0.61	0.62	0.61	0.59	0.60	0.60	0.62	0.60	0.58
Logistic-HAR-KS	0.67	0.67	0.68	0.67	0.65	0.66	0.67	0.68	0.66	0.65
Panel B: Machine learning forecasting model										
RF-HAR-KS	0.72	0.72	0.73	0.72	0.72	0.72	0.72	0.73	0.72	0.72
LightGBM-HAR-KS	0.73	0.73	0.73	0.73	0.73	0.73	0.73	0.73	0.73	0.72
XGBoost-HAR-KS	0.72	0.73	0.72	0.73	0.72	0.72	0.73	0.72	0.73	0.72
Panel C: Deep learning forecasting model										
FFN-HAR-KS	0.73	0.73	0.73	0.74	0.74	0.73	0.73	0.73	0.74	0.74
CNN-HAR-KS	**0.76**	**0.77**	**0.76**	**0.76**	**0.75**	**0.76**	**0.76**	**0.76**	**0.76**	**0.75**

Note: The bold numbers in the table indicate that the model yields the highest forecasting accuracy.

substantial significant first-order autocorrelation ACF (1) for all HAR-type components except for SJ, indicating high persistence. The Ljung–Box test at both 5 and 21 lags confirms the high persistence of all HAR-type components.

Fig. 1 depicts the time series plots of the HAR-type indicators from Table 1. During the stock market turmoil of 2015 and the COVID-19 pandemic in 2020, we observe significant increases in RV (Fig. 1a).[3] Furthermore, as shown in Fig. 1c, a substantial number of jumps

[3] The study of China's stock market fluctuations in 2015 mirrors the global financial crisis that occurred in 2008. During the fluctuations (including signed jumps, which initially exhibited positive jumps followed by negative jumps) experienced by China's stock market in 2015, there were significant signed jumps in the prices of individual stocks, and the stock market's risk exposure was substantial, resulting in significant losses for investors. Analyzing China's stock market volatility in 2015 is a representative approach.

occurred during these periods. Similarly, RS− (RV using negative 5-min returns) increases significantly during the 2015 market turbulence and 2020 pandemic, whereas RS+ (RV using positive 5-min returns) only shows an increase in 2015. This finding suggests that stock market declines were the primary driver of volatility during the 2020 pandemic. Fig. 1g, Fig. 1h, and Fig. 1i show the trend in signed jumps (RS+ minus RS−). We observe that the signed jump (SJ) indicator initially exhibited positive jumps followed by negative jumps during the 2015 market turbulence. In contrast, during the COVID-19 pandemic in 2020, the SJ indicator primarily showed negative jumps. These findings underscore the distinct characteristics of the two high-volatility periods and emphasize the importance of using multidimensional HAR-type measures to model real volatility more effectively.

4. Empirical application

4.1. *Out-of-sample forecast results for CSI 300 index volatility*

Table 4 reports the out-of-sample forecast results for the CSI 300 index volatility direction using the full testing sample (1000 days) and four subsamples (250 days). First, the logistic-HAR-KS, including all HAR components, outperforms other traditional HAR-type models in out-of-sample volatility forecasting. The conventional HAR-type models with a jump component can outperform other traditional HAR-type models (Andersen et al., 2007; Patton & Sheppard, 2015). Subsequently, as expected, we find that ML models (RF-HAR-KS, LightGBM-HAR-KS, and XGBoost-HAR-KS models) can produce more accurate volatility prediction results than traditional HAR-type models. This finding emphasizes how ML outperforms conventional HAR-type models in volatility predictions (Christensen et al., 2023). Finally, DL models (FFN-HAR-KS and CNN-HAR-KS models) consistently outperform ML models in terms of out-of-sample volatility forecasting. Moreover, the CNN-HAR-KS model outperforms all other models in various samples, demonstrating the critical importance of automated signal generation in the form of HAR-type images for volatility forecasting.

4.2. *Out-of-sample forecast results at individual stock level*

Table 5 shows the out-of-sample forecast results for the direction of individual stock volatility using the same methodology as in Table 4. Within traditional logistic-HAR-type models, the logistic-HAR-KS model, incorporating all HAR-type components, has the highest predictive accuracy of 67 %, outperforming any specific HAR-type model. When compared with traditional logistic models, ML-based HAR-KS models achieve a higher level of forecasting precision (approximately 72 %), whereas the CNN-HAR-KS model, using image-based learning, achieves a prediction accuracy of approximately 76 %. The CNN-HAR-KS model effectively extracts HAR component information at the individual stock level, resulting in superior predictive performance.

4.3. *Robust check*

4.3.1. *Alternative forecast evaluations*

We apply alternative metrics to check the robustness of our results. Specifically, we introduce the sensitivity, specificity, and Youden index.

$$Sensitivity = \frac{TP}{TP + FN}, \tag{17}$$

$$Specificity = \frac{TN}{FP + TN}, \tag{18}$$

$$Youden's\ Index = Sensitivity + Specificity - 1, \tag{19}$$

where *TP* means true positive rate, *TN* denotes True Negative rate, *FN* indicates false negative rate, and *FP* means false positive rate. That is, sensitivity is the probability of a positive test result, conditioned on the individual being truly positive. Specificity is the probability of a negative test result, conditioned on the individual being truly negative. Finally, Youden's index summarizes a test's accuracy, balancing its ability to correctly identify positive and negative cases.

Table 6
Alternative forecast evaluations.

	Accuracy	AUC	Sensitivity	Specificity	Youden's Index
Panel A: HAR family logistic forecasting model					
Logistic-HAR	0.63	0.64	0.64	0.63	0.27
Logistic-HARJ	0.64	0.64	0.68	0.59	0.27
Logistic-HARCJ	0.63	0.64	0.59	0.67	0.26
Logistic-HARRSI	0.63	0.63	0.57	0.68	0.25
Logistic-HARRSII	0.64	0.64	0.64	0.63	0.28
Logistic-HARSJI	0.59	0.59	0.55	0.62	0.17
Logistic-HARSJII	0.63	0.63	0.63	0.63	0.25
Logistic-HAR-KS	0.65	0.65	0.68	0.61	0.30
Panel B: Machine learning forecasting model					
RF-HAR-KS	0.74	0.74	0.71	**0.77**	0.48
Light GBM-HAR-KS	0.75	0.75	0.75	0.75	0.5
XGBoost-HAR-KS	0.74	0.74	0.72	0.76	0.48
Panel C: Deep learning forecasting model					
FFN-HAR-KS	0.75	0.75	0.77	0.74	0.51
CNN-HAR-KS	**0.78**	**0.78**	**0.81**	0.75	**0.56**

Note: The bold numbers in the table indicate that the model yields the highest forecasting accuracy.

Table 6 reports forecasting accuracy using alternative forecast measures. We include accuracy and AUC for comparison. Notably, the CNN-HAR-KS model still demonstrates the best prediction performance, achieving the highest sensitivity, and the Youden index indicates the strongest overall predictive capability. Although the RF-HAR-KS model

Table 7
Alternative parameter.

	Accuracy	AUC	Sensitivity	Specificity	Youden's Index
Panel A: HAR family logistic forecasting model					
Logistic-HAR	0.63	0.64	0.64	0.63	0.27
Logistic-HARJ	0.64	0.64	0.68	0.59	0.27
Logistic-HARCJ	0.63	0.64	0.59	0.67	0.26
Logistic-HARRSI	0.63	0.63	0.57	0.68	0.25
Logistic-HARRSII	0.64	0.64	0.64	0.63	0.28
Logistic-HARSJI	0.59	0.59	0.55	0.62	0.17
Logistic-HARSJII	0.63	0.63	0.63	0.63	0.25
Logistic-HAR-KS	0.65	0.65	0.68	0.61	0.30
Panel B: Machine learning forecasting model					
RF-HAR-KS	0.74	0.74	0.71	**0.77**	0.48
Light GBM-HAR-KS	0.75	0.75	0.75	0.75	0.5
XGBoost-HAR-KS	0.74	0.74	0.72	0.76	0.48
Panel C: Deep learning forecasting model					
FFN-HAR-KS	0.75	0.75	0.77	0.74	0.51
CNN-HAR-KS	**0.78**	**0.78**	**0.82**	0.74	**0.56**

Note: The bold numbers in the table indicate that the model yields the highest forecasting accuracy.

Table 8
Alternative test lengths.

	750 days testing size					500 days testing size				
	Accuracy	AUC	Sensitivity	Specificity	Youden	Accuracy	AUC	Sensitivity	Specificity	Youden
Panel A: HAR family logistic forecasting model										
Logistic-HAR	0.65	0.65	0.72	0.57	0.29	0.66	0.67	0.91	0.43	0.34
Logistic-HARJ	0.65	0.66	0.76	0.54	0.30	0.64	0.66	**0.93**	0.38	0.31
Logistic-HARCJ	0.65	0.66	0.64	0.66	0.30	0.67	0.67	0.80	0.54	0.34
Logistic-HARRSI	0.64	0.64	0.63	0.64	0.27	0.67	0.67	0.83	0.51	0.34
Logistic-HARRSII	0.65	0.65	0.71	0.58	0.30	0.64	0.64	0.85	0.51	0.29
Logistic-HARSJI	0.60	0.61	0.60	0.61	0.21	0.62	0.63	0.77	0.48	0.25
Logistic-HARSJII	0.64	0.65	0.69	0.60	0.28	0.64	0.65	0.85	0.45	0.30
Logistic-HAR-KS	0.66	0.66	0.73	0.59	0.32	0.65	0.66	0.74	0.57	0.31
Panel B: Machine learning forecasting model										
RF-HAR-KS	0.75	0.75	0.72	**0.78**	0.49	0.74	0.74	0.72	**0.77**	0.49
LightGBM-HAR-KS	0.76	0.76	0.75	0.77	0.51	0.75	0.75	0.76	0.75	0.51
XGBoost-HAR-KS	0.76	0.76	0.73	0.76	0.49	0.76	0.76	0.76	0.75	0.52
Panel C: Deep learning forecasting model										
FFN-HAR-KS	0.76	0.76	0.77	0.75	0.52	0.76	0.76	0.76	0.75	0.51
CNN-HAR-KS	**0.78**	**0.78**	**0.81**	0.75	**0.57**	**0.78**	**0.78**	0.81	0.74	**0.55**

Note: The bold numbers in the table indicate that the model yields the highest forecasting accuracy.

has the highest specificity, its ability to predict true positives (i.e., volatility decreases in one direction) is only 71 %, thus making its overall performance inferior to that of the CNN-HAR-KS model. In summary, the proposed model's predictive power is robust when compared with alternative forecast measures.

4.3.2. *Alternative parameter*

Parameter selection plays a crucial role in determining the generalizability of neural network models. In the main results, we identify the optimal parameters by evaluating the loss of the validation set. To show the robustness of our results, we randomly choose a set of parameters for prediction (Table 7).[4] The results reveal that the proposed CNN-HAR-KS model outperforms the other models. Furthermore, the findings are robust across different parameters.

4.3.3. *Alternative test length*

The length of the test sample significantly affects predictive performance. The main results reveal that the training and testing samples are approximately divided as 60:40 with a 1000-day testing size. To ensure the robustness of the results, we apply the last 750 and 500 days (approximately 70:30 and 80:20, respectively) as test sets for prediction. Table 8 presents the forecast results. Unsurprisingly, the CNN-HAR-KS model achieves the highest accuracy, AUC score, and Youden index. Therefore, our findings are robust to alternative test lengths.

4.3.4. *Recursive forecasting window*

We use a rolling window approach to generate forecasts in the main results. We also use a recursive forecasting window approach to estimate model parameters and generate out-of-sample forecasts further to evaluate the CNN-HAR-KS model's robust predictive power. Table 9 presents the predicted results. The proposed CNN-HAR-KS model maintains dominant performance, indicating robustness within the recursive forecasting window.

Table 9
Recursive forecasting window.

	Accuracy	AUC	Sensitivity	Specificity	Youden's Index
Panel A: HAR family logistic forecasting model					
Logistic-HAR	0.61	0.61	0.35	0.86	0.21
Logistic-HARJ	0.63	0.63	42.00	84.00	26.00
Logistic-HARCJ	0.61	0.60	35.00	85.00	20.00
Logistic-HARRSI	0.59	0.58	27.00	**90.00**	17.00
Logistic-HARRSII	0.62	0.62	40.00	84.00	23.00
Logistic-HARSJI	0.59	0.58	31.00	80.00	11.00
Logistic-HARSJII	0.62	0.62	37.00	83.00	19.00
Logistic-HAR-KS	0.64	0.64	0.64	0.63	0.27
Panel B: Machine learning forecasting model					
RF-HAR-KS	0.75	0.75	72.00	77.00	49.00
Light GBM-HAR-KS	0.75	0.75	76.00	74.00	49.00
XGBoost-HAR-KS	0.74	0.74	72.00	76.00	48.00
Panel C: Deep learning forecasting model					
FFN-HAR-KS	0.75	0.75	**88.00**	63.00	50.00
CNN-HAR-KS	**0.78**	**0.78**	82.00	73.00	**56.00**

Note: The bold numbers in the table indicate that the model yields the highest forecast accuracy.

5. Portfolio exercise

Financial professionals may deliberate on the extent to which the statistical forecasting advantages offered by our models translate into tangible economic benefits. To examine the economic significance of our predictive models' forecasts of RV direction (RVD), we conduct a comprehensive analysis based on the concept of the low volatility anomaly, also known as the idiosyncratic volatility puzzle. It has garnered significant attention in asset pricing. Based on US stock market data, Ang et al. (2006) discovered that stocks with higher idiosyncratic volatility in a given month tend to exhibit lower returns in the subsequent month. Furthermore, classical pricing factors do not explain this phenomenon. Guo and Savickas (2010) and Chen and Petkova (2012) conduct extensive empirical research, demonstrating the presence of an idiosyncratic volatility anomaly in the US market using a variety of robustness tests. Moreover, Gu et al. (2018) confirmed the existence of a unique volatility anomaly in the Chinese A-share market. These findings

[4] We randomly choose one of the parameter sets described in Appendix C, characterized by "batch_size = 8," "dropout = 0.3," "number of CNN layers = 1," and "L2 rate = 0.001."

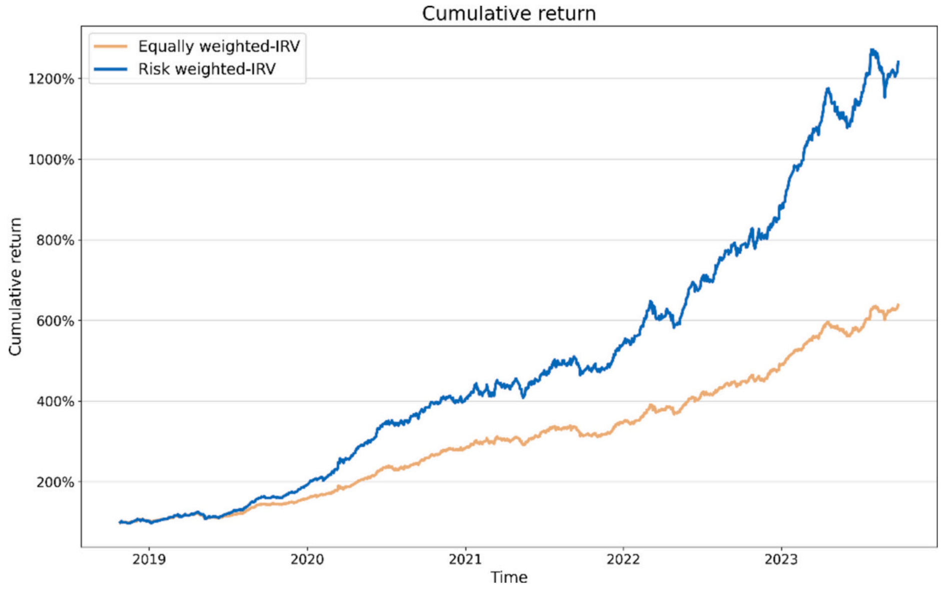

Fig. 2. Cumulative return with Invest on the Low Realized Volatility (ILRV) investment strategy.

establish a theoretical foundation for developing investment strategies based on volatility forecasting. Consequently, we employ the “Invest on the Low Realized Volatility” (ILRV) investment strategy, which involves holding a long position in individual stocks with low RV and a short position in stocks with a high RV. The cumulative return may be defined as follows:

$$ilrvr_{i,t} = \begin{cases} 1 + r_{i,t}, & if \quad \widehat{RV_{i,t}} - RV_{i,t-1} < 0, \\ 1 - r_{i,t}, & if \quad \widehat{RV_{i,t}} - RV_{i,t-1} \geq 0, \end{cases} \tag{20}$$

where $ilrvr_{i,t}$ denotes the return based on the ILRV strategy, $r_{i,t}$ represents the stock i return at time t, $\widehat{RV_{i,t}}$ indicates the forecasts for stock i at time t, and $RV_{i,t-1}$ denotes the true RV for stock i at time $t-1$.

Then, we consider all individual stocks as assets to construct investment portfolios. We explore two allocation methods, namely, equal weighted (EW) and risk weighted (RW). The portfolio returns can be defined as follows for the EW allocation approach.

$$r_t^{EW} = \frac{1}{N}\sum_{i=1}^{N} ilrvr_{i,t}, \tag{21}$$

where r_t^{EW} means the equal average of $ilrvr_{i,t}$ based on the ILRV strategy and N is the number of stocks. Similarly, the portfolio returns with RW allocation are as follows:

$$r_t^{RW} = \sum_{i=1}^{N} w_{i,t} ilrvr_{i,t}, \tag{22}$$

$$w_{i,t} = \frac{\left|\widehat{RV_{i,t}} - RV_{i,t-1}\right|}{\sum_{j=0}^{5}\left|\widehat{RV_{i,t-j}} - RV_{i,t-j-1}\right|}, \tag{23}$$

where r_t^{RW} represents the weighted average of $ilrvr_{i,t}$ based on the ILRV strategy. $w_{i,t}$ denotes the risk weights for stock i. That is, we provide the higher weights to more volatile stocks for the higher excess returns.

We start by assuming perfect forecasting of the RV direction and calculating cumulative returns from August 20, 2019, to September 28, 2023. Fig. 2 illustrates the cumulative returns for the RW-ILRV and EW-ILRV strategies. When we apply the ILRV strategy, we observe substantial arbitrage space on the Chinese stock market. Both the RW-ILRV and EW-ILRV strategies show an upward trend in cumulative returns, indicating the possibility of positive returns on investment using an ILRV-based approach. In particular, the RW-ILRV strategy generates a higher cumulative value than the EW-ILRV strategy, implying that investing in riskier assets can lead to higher returns.

Table 10
IRV investment strategy performance.

	RF-HAR-KS	LightGBM-HAR-KS	XGBoost-HAR-KS	FFN-HAR-KS	CNN-HAR-KS
Panel A: Descriptive statistics					
N	1000	1000	1000	1000	1000
Mean (%)	−0.032	−0.007	−0.007	0.025	**0.078**
STD (%)	**1.767**	1.781	1.778	1.806	1.803
Max (%)	6.677	6.777	7.212	6.947	**7.232**
Min (%)	**−11.119**	−11.205	−11.204	−11.211	−11.216
Panel B: Sharpe ratio and information ratio					
Sharpe ratio	−0.018	−0.004	−0.004	0.014	**0.043**
IR	−0.030	−0.004	−0.004	0.028	**0.078**
Panel C: Max drawdown, average drawdown, and max 1-day loss					
Avg DD	−0.265	−0.258	−0.254	−0.233	**−0.182**
Max DD	−0.747	−0.728	−0.720	−0.703	**−0.615**
Max 1D loss	**−0.111**	−0.112	−0.112	−0.112	−0.112

Note: Max drawdown (Max DD), average drawdown (Avg DD), and maximum 1-day loss (Max 1D loss) denote different types of drawdown. The lower values of the Max DD, Avg DD, and Max 1D loss indicate the lower downside risk over a specified period.

As a result, our asset allocation method is the RW-ILRV investment strategy. To be more specific, we buy all individual stocks with expected volatility decreases and sell all individual stocks with anticipated volatility increases. The magnitude of their changes, denoted as $\omega_{i,t}$, serves as the trading weight for each stock. Table 10 compares the investing utility of the RF-HAR-KS, LightGBM-HAR-KS, XGBoost-HAR-KS, FFN-HAR-KS, and CNN-HAR-KS models using the RW-ILRV strategy.[5] First,

[5] Given the Chinese stock market’s short-selling limits, we calculate the cumulative only taking long positions using the RW-ILRV strategy. The CSI 300 index return serves as the benchmark for calculating the information ratio (IR).

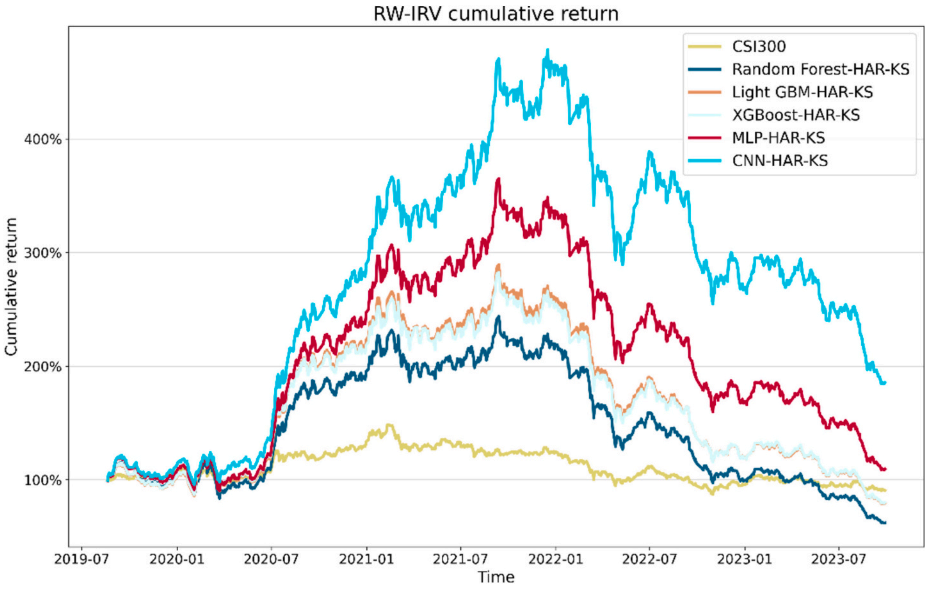

Fig. 3. Cumulative return with risk weighted-Invest on the Low Realized Volatility (RW-ILRV) strategy.

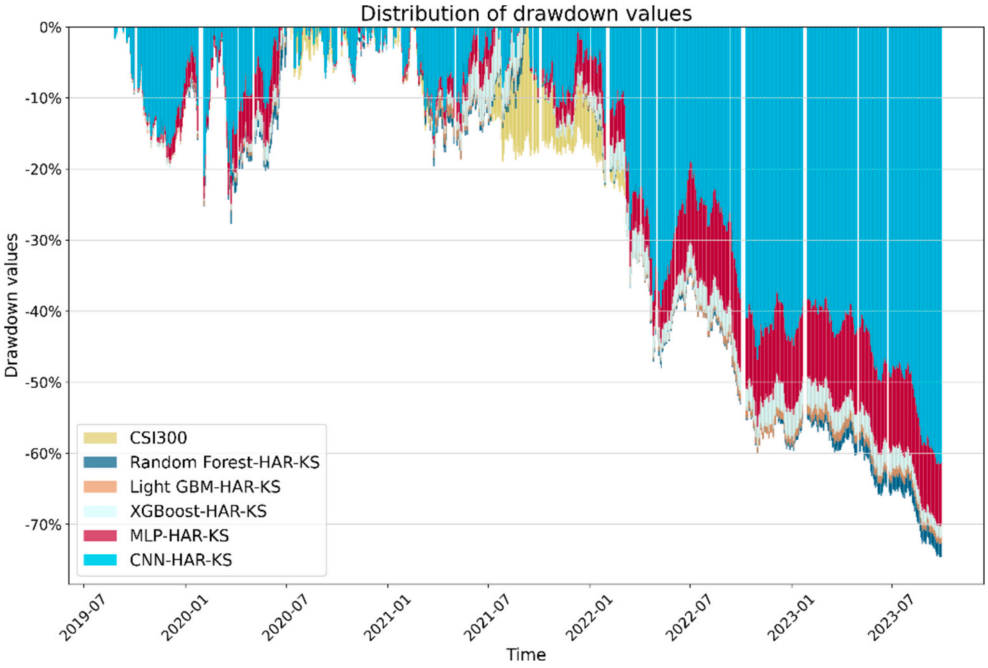

Fig. 4. Distribution of drawdown values with the risk weighted-Invest on the Low Realized Volatility (RW-ILRV) investment strategy

the CNN-HAR-KS model produces the highest mean (0.078 %) and maximum (7.232 %) investment returns, outperforming all other proposed models. The FFN-HAR-KS model achieves the second-highest average return, trailing only the remaining ML-based models. Panel B also includes the Sharpe ratio and information ratio (IR). It is worth noting that the DL-based HAR-type models (FFN-HAR-KS and CNN-HAR-KS) have higher Sharpe ratios and IR values than the ML-based HAR-type models. Importantly, the proposed CNN-HAR-KS model demonstrates the largest Sharpe ratio and IR compared to the FFN-HAR-KS model, indicating its superiority in extracting information from image-form HAR-type features. Panel C displays the drawdown statistics for all models examined. Interestingly, the CNN-HAR-KS model achieves the smallest average and maximum drawdown values, indicating superior risk management capability.[6]

Fig. 3 depicts the cumulative returns of the models based on percentage points; if the forecasting model yields a higher percentage point value than the other models considered, it suggests that the forecasting model may provide some excess returns to investors. The CNN-HAR-KS model (blue line) has the highest cumulative return (186 %), outperforming all other models. Notably, it outperforms the second-best

[6] Drawdown is the maximum observed loss from a portfolio's peak to trough before reaching a new peak. It serves as an indicator of the potential loss, with a lower drawdown indicating a reduced risk profile.

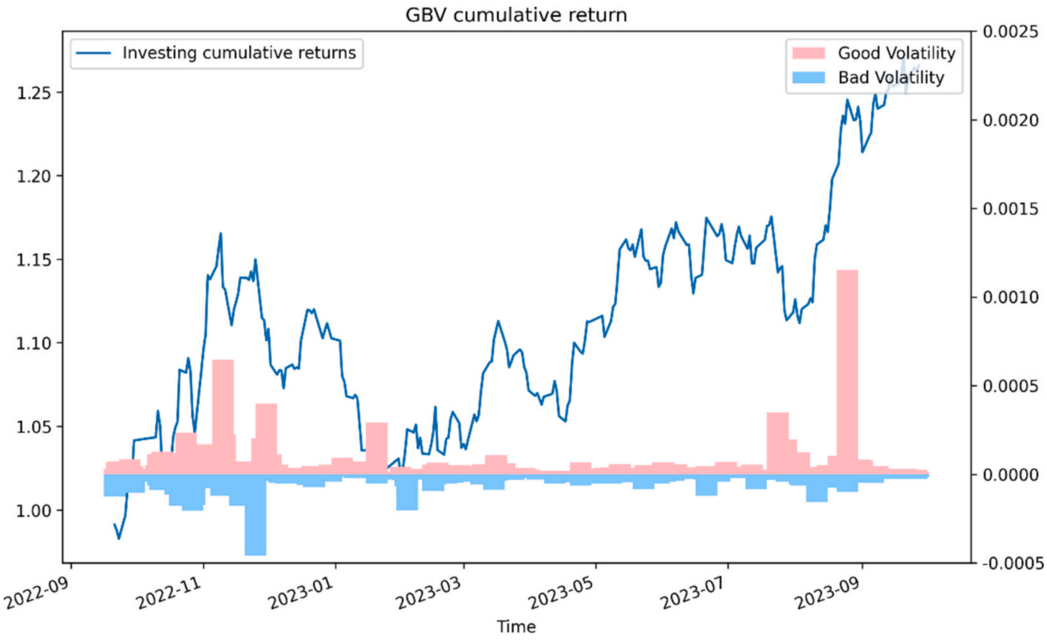

Fig. 5. Investment utility with good/bad volatility predictions.

FFN-HAR-KS model by 70 %. Meanwhile, the ML-based HAR-KS models, such as RF-HAR-KS, LightGBM-HAR-KS, and XGBoost-HAR-KS, can generate some excess returns during a positive performance in the CSI 300 index before February 2021. However, as the index weakened, the returns of these models gradually declined, thus resulting in losses.

Fig. 4 depicts the drawdown values of all models over time. Throughout the observed period, the DL-based models show only minor drawdowns when compared to the CSI 300 index benchmark. Specifically, during the phase of significant retracement of the CSI 300 index (January 2021 to October 2023), the shaded area, representing the drawdown of the CNN-HAR-KS model, remains substantially smaller than that of the index or other considered models (indicated by red and blue areas). This finding demonstrates that the CNN-HAR-KS model achieves significantly lower drawdown values in volatile market environments. In summary, the CNN-HAR-KS model's RV forecasts show the potential for greater economic value than the other models under consideration.

6. Exploration of volatility classification prediction

After reviewing the existing literature in the field of financial forecasting, one significant and promising branch involves extracting various patterns in time series of asset prices, which is equivalently classified, and further investigating their predictability for variables of interest across different market scenarios (Jiang et al., 2023; Yun et al., 2021). In recent years, certain scholars have attempted to incorporate prior knowledge of price classification into volatility forecasting. Volatility is classified into good and bad components based on the sign of asset price returns (BenSaida, 2021; Bollerslev et al., 2020; Patton & Sheppard, 2015; Zhang & Zhao, 2023). Unfortunately, to the best of our knowledge, the predictability of the volatility classification remains underexplored. Therefore, this study proposes a pilot study on applying volatility classification in volatility forecasting.

As shown in the previous section, the proposed CNN-HAR-KS model effectively predicts the direction of volatility. To validate the proposed model's robustness and expandability, we examine its performance in existing volatility forecasting classification problems, such as predicting GV and BV or identifying negative jumps. Theoretically, the CNN-HAR-KS model includes various HAR-type components, improving its predictive capability for other classification problems. We first predict good and bad volatility using the definitions of Patton and Sheppard (2015). The predictive target is defined as follows:

$$GBV = \begin{cases} 1, & if \quad GV \geq BV, \\ 0, & if \quad GV < BV, \end{cases} \tag{24}$$

where GV and BV denote good and bad volatility, respectively. If good volatility is greater than bad volatility, we classify it as 1; otherwise, we classify it as 0.[7] By doing so, we follow a simple investment strategy: take long positions when good volatility outnumbers bad volatility.

Fig. 5 presents the investment utility. The red bars represent good volatility, while the blue bars represent bad volatility, with the total representing RV. We observe that we can achieve an approximate cumulative return of 26 % by predicting good and bad volatility. This indicates that the proposed CNN-HAR-KS model equally applies to other RV classification problems.

We also use the proposed CNN-HAR-KS model to predict negative jumps using Patton and Sheppard's (2015) SJ− measure. This limits investors' losses during market downturns. Fig. 6 shows the forecasting results. Despite certain prediction errors, we determine that the CNN-HAR-KS model can effectively predict the occurrence of negative jumps (e.g., in August 2023). Therefore, our model is applicable to RVD classification problems and demonstrates generalizability to other RV classification issues, thereby enriching the exploration of RV-related classification problems.

7. Conclusions

This study proposes CNN-HAR-KS, a novel CNN model that uses image-based HAR-type components to predict the direction of stock market volatility. The CNN-HAR-KS model's main advantage is its ability to leverage CNNs' strengths in image classification tasks. By transforming HAR-type components into feature matrices represented as two-dimensional images, the model takes advantage of CNNs' superior image classification capabilities, thus resulting in improved volatility forecasting accuracy.

[7] Good volatility can be written as $RS^{+} = \sum_{i=1}^{n} r_i^2 I\{r > 0\}$, bad volatility is presented as $RS^{-} = \sum_{i=1}^{n} r_i^2 I\{r < 0\}$.

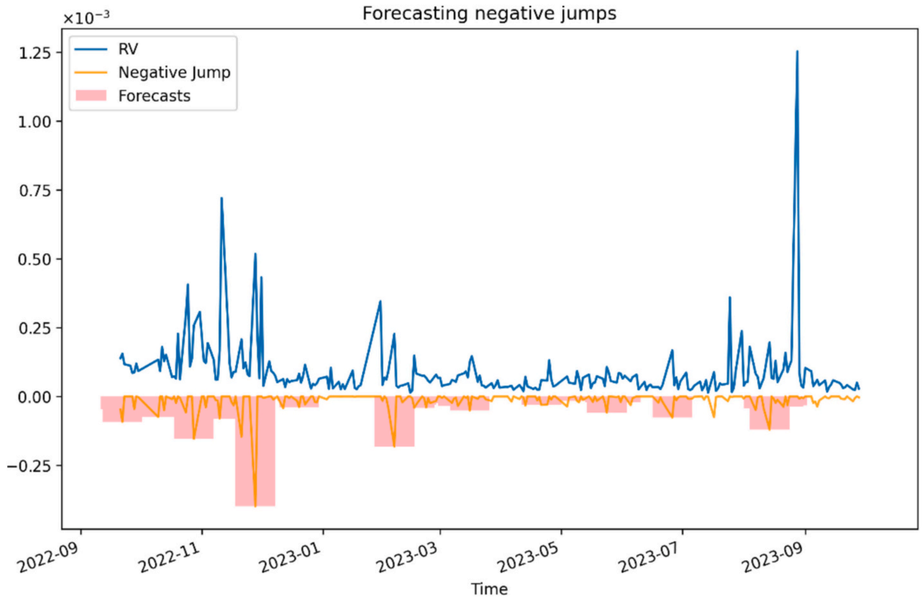

Fig. 6. Forecasting negative jumps.

When applied empirically to China's stock market, the CNN-HAR-KS model outperforms other models in terms of forecasting performance. Furthermore, this superior performance is robust across various evaluations, parameters, and test sizes. The Sharpe ratio, IR, and minor drawdowns all contribute to the CNN-HAR-KS model's economic value. These results underscore the significance of using image-formatted HAR-type components as model inputs.

This study has significant theoretical importance and provides significant economic benefits. From a theoretical standpoint, this study uses the advantages of the CNN model in processing 2-dimensional data (i.e., images) to transform time series data into images, capturing the typical characteristics of time series data (i.e., nonlinearity) from a high-dimensional perspective. Moreover, this framework can be applied to other RV-related classification problems, adding theoretical value to the literature on volatility classification prediction. Furthermore, the proposed CNN-HAR-KS model fully utilizes information from HAR-type components, resulting in economic interpretability comparable to that of HAR-type logit models. From a practical standpoint, the CNN-HAR-KS model can provide greater economic value to investors through RV direction or good/bad volatility as well as by predicting negative jumps, making it useful in risk management.

Our findings open up new avenues for future research into RV classification problems. Future research could expand the CNN-HAR-KS model to address other RV-related classification issues, such as categorizing jumps into more than two categories and predicting their specific intensity. Furthermore, RV could be divided into more granular categories, allowing for easier integration with value-at-risk management strategies.

CRediT authorship contribution statement

Nan Hu: Conceptualization, Methodology, Software, Data curation, Validation, Formal analysis, Investigation, Resources, Writing – original draft, Writing – review & editing. **Xuebao Yin:** Methodology, Funding acquisition, Investigation, Resources, Writing – review & editing. **Yuhang Yao:** Conceptualization, Investigation, Funding acquisition, Visualization, Writing – original draft.

Declaration of competing interest

We don't use AI in our writing, and no potential conflict of interest was reported by the authors.

Acknowledgment

This work was supported by the Anhui Provincial Quality Engineering Project under Grant No. 2022xscx083.

Appendix A. The specification of LightGBM

LightGBM seeks to find an approximation $\widehat{f}(x)$ to a specific function $f^*(x)$ that minimizes the expected value of a given loss function $L(y,f(x))$ given the supervised training set $X = \{(x_i,y_i)\}_{i=1}^n$.

$$\widehat{f} = \underset{f}{argmax} E_{y,X} L(y,f(x)), \tag{A.1}$$

LightGBM integrates a number of T regression trees $\sum_{t=1}^T f_t(X)$ to approximate the final model, which is as follows:

$$f_T(X) = \sum_{t=1}^T f_t(X). \tag{A.2}$$

The expression $w_{q(x),q\in\{1,2,\ldots,J\}}$ could be used to represent the regression trees, where J represents the number of leaves, q represents the tree's decision rules, and w is a vector representing the sample weight of leaf nodes. Therefore, at step t, LightGBM will be trained in an additive form as follows:

$$\Gamma_t = \sum_{i=1}^{n} L(y_i, F_{t-1}(x_i) + f_t(x_i)\,). \tag{A.3}$$

Newton's method is used in LightGBM to quickly approximate the objective function. For simplicity, the constant term in (13) can be removed, and the equation becomes

$$\Gamma_t \cong \sum_{i=1}^{n} \left(g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)\, \right), \tag{A.4}$$

where the loss function's first- and second-order gradient statistics are indicated by the variables g_t and h_t. Let I_j represent the leaf j sample set. Then, we can transform (14) in the following way:

$$\Gamma_t = \sum_{i=1}^{j} \left(\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right). \tag{A.5}$$

The optimal weight scores of each leaf node w_j^* and the extreme value of Γ_K for a given tree structure $q(x)$ can be solved as follows:

$$\Gamma_T^* = -\frac{1}{2} \sum_{i=1}^{J} \frac{\left(\sum_{i \in I_j} g_i \right)^2}{\sum_{i \in I_j} h_i + \lambda}, \tag{A.6}$$

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}, \tag{A.7}$$

where the scoring function Γ_T^* may be thought of as evaluating the quality of the tree structure q. Finally, the objective function after adding the split is as follows:

$$\Gamma_T^* = -\frac{1}{2} \left(\frac{\left(\sum_{i \in I_L} g_i \right)^2}{\sum_{i \in, I_L} h_i + \lambda} + \frac{\left(\sum_{i \in I_R} g_i \right)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{\left(\sum_{i \in I} g_i \right)^2}{\sum_{i \in I} h_i + \lambda} \right), \tag{A.8}$$

where the sample sets for the left and right branches are denoted by I_L and I_R, respectively. LightGBM is an efficient way to handle large amounts of data and features because, unlike other algorithms, it grows trees vertically rather than horizontally. This makes it unique from other classic GBDT-based techniques like XGBoost and GBDT.

Generally, the forecasting method accuracy would be significantly influenced by the hyperparameters. Thus, before using LightGBM, we must first determine the number and range of variation in its hyperparameters.

Appendix B. The specification of XGBoost

Statistically, the XGBoost model can be expressed as follows.

$$\widehat{y} = \widehat{f}(x) = \sum_{i=0}^{K} \widehat{f}_i(x), \quad \widehat{f}_i(x) \in F \tag{B.1}$$

where $F = f(x) = w_{q(x)} \left(q : \mathbb{R}^m \rightarrow T, w \in \mathbb{R}^T \right)$,$q$denotes the tree structure that facilitates input mapping to the corresponding leaf index. T represents the total number of leaves within the tree, while w_i signifies the score associated with the i-th leaf. To acquire the set of functions used in the model, our goal is to minimize the following loss function:

$$\mathscr{L}(\phi) = \sum_i l(\widehat{y}_i, y_i) + \sum_k \Omega(f_k), \tag{B.2}$$

$$\Omega(f_k) = \gamma T + \frac{1}{2} \lambda \|w\|^2. \tag{B.3}$$

l is a differentiable convex loss function that measures the difference between the predicted y_i and target $\widehat{y}_i$. The $\sum_k \Omega(f_k)$ penalizes the model's complexity, which helps smooth the final learned weights to avoid overfitting.

For the t-th iteration, f_t is added to minimize the objective function.

$$\mathscr{L}^{(t)} = \sum_i l\left(\widehat{y}_i^{t-1} + f_t(x_i)\,, y_i\right) + \sum_t \Omega(f_k). \tag{B.4}$$

Using a Taylor series expansion, we can obtain the second-order approximation of the objective function as follows:

$$\mathscr{L}^{(t)} = \sum_{i}^{n}\left[l\left(\widehat{y}_i^{t-1}, y_i\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_i^2(x_i)\right] + \Omega(f_k) \tag{B.5}$$

$$where \quad g_i = \partial_{\widehat{y}_i t-1} l\left(\widehat{y}_i^{t-1}, y_i\right) \quad and \quad h_i = \partial^2_{\widehat{y}_i t-1} l\left(\widehat{y}_i^{t-1}, y_i\right). \tag{B.6}$$

Then, if we let the $I_j = \{i | q(x_i) = j\}$ be the instance set of leaf j and remove the constant terms, the objective function is rewritten as follows:

$$\mathscr{L}^{(t)} = \sum_{i}^{n}\left[g_i f_t(x_i) + \frac{1}{2} h_i f_i^2(x_i)\right] + \gamma T + \frac{1}{2}\lambda \|w\|^2 + \sum_{j=1}^{T}\left[\left(\sum_{i\in I_j} g_i\right) w_i + \frac{1}{2}\left(\sum_{i\in I_j} h_i + \lambda\right) w_j^2\right] + \gamma T. \tag{B.7}$$

Given a fixed tree structure, determining the optimal weight for leaf *j*requires differentiating this equation concerning w and setting it equal to zero, allowing for the calculation of the optimal weight.

$$w_j^* = -\frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda}. \tag{B.8}$$

Then, by replacing w_j^*, the optimal value for the given tree structure would be:

$$\mathscr{L}^{(t)}(q) = -\frac{1}{2}\sum_{j=1}^{T}\frac{\left(\sum_{i\in I_j} g_i\right)^2}{\sum_{i\in I_j} h_i + \lambda} + \gamma T. \tag{B.9}$$

Appendix C. Model parameter setting

Models	Parameters	Search Space
FFN-HAR-KS	Activation functions in the input layer	Relu
	Activation functions in the output layer	Sigmoid
	Batch size	[8, 16]
	Dropout	[0.3, 0.7]
	L2 rate	[0.001]
	learning rate (lr)	Reduce LR On Plateau, *min lr = 0.0001*
	Number of epochs	The maximum number of epochs is 3000
CNN-HAR-KS	Activation functions in the convolutional layer	Relu
	Number of CNN layers	[1,2]
	Activation functions in the output layer	*Softmax*
	Batch size	[8, 16]
	Dropout	[0.3, 0.7]
	L2 rate	[0.001]
	learning rate (lr)	Reduce LR On Plateau, *min lr = 0.0001*
	Number of epochs	The maximum number of epochs is 3000

Appendix D. Financial volatility and trading-related abbreviations

1. Volatility Measures

RV (Realized Volatility): A historical volatility measure computed using high-frequency data, providing an ex-post measure of return variation.

BPV (Bi-Power Variation): A robust estimator of integrated variance that is immune to jumps, used to separate continuous sample paths from jumps.

TQ (Tri-Power Quarticity): A robust estimator used to compute the asymptotic standard errors of realized volatility measures.

2. Jump Detection and Decomposition

ABD Jump/CSP: Jump component and Continuous Sample Path identified using the Andersen-Bollerslev-Dobrev jump test methodology.

BNS Jump/CSP: Jump component and Continuous Sample Path detected using the Barndorff-Nielsen-Shephard jump test methodology.

Jo Jump/CSP: Jump component and Continuous Sample Path determined using the Jo jump test methodology.

3. Return Decomposition

RS$^+$ (Realized Semi-Variance $^+$): Realized variance computed using positive 5-min returns.

RS$^-$ (Realized Semi-Variance $^-$): Realized variance computed using negative 5-min returns.

Negative RV: Realized volatility multiplied by an indicator function for negative daily returns.

SJ (Signed Jump): The difference between RS$^+$ and RS$^-$, representing signed jump variation.

SJ$^+$/SJ$^-$: Signed jump variation decomposed into positive and negative components.

4. Predictive Models

4.1 Logistic Regression Series

Logistic-HAR: The logistic regression model that includes the daily, weekly, and monthly averages of RV.

Logistic-HARJ: The Logistic-HAR model augmented with the daily jump component.

(*continued on next page*)

(*continued*)

Logistic-HARCJ: The logistic regression model incorporating the daily, weekly, and monthly averages of both the continuous sample path and the jump components.

Logistic-HARRSI: The logistic regression model including the daily RS^{+} and RS^{-}, as well as the weekly and monthly averages of RV.

Logistic-HARRSII: The Logistic-HARRSI model with an additional term that interacts the lagged RV with an indicator function for negative daily returns.

Logistic-HARSJI: The logistic regression model that includes the daily signed jump (SJ), bi-power variation (BPV), and the weekly and monthly averages of RV.

Logistic-HARSJII: The Logistic-HARSJI model that further decomposes the signed jump variation into positive and negative jumps.

Logistic-HAR-KS: The logistic regression model that includes all HAR components (the "Kitchen Sink" model).

4.2 Machine Learning Models

RF-HAR-KS: A Random Forest model incorporating all HAR-type components.

LightGBM-HAR-KS: A Light Gradient Boosting Machine model with all HAR components.

XGBoost-HAR-KS: An Extreme Gradient Boosting model with all HAR components.

4.3 Deep Learning Models

FFN-HAR-KS: A Feed-Forward Neural Network model with all HAR-type components.

CNN-HAR-KS: A Convolutional Neural Network model with all HAR components.

5. Market-Related Terms

CSI 300: The China Securities Index 300, a capitalization-weighted stock market index tracking the performance of 300 stocks traded on the Shanghai and Shenzhen stock exchanges.

ILRV (Invest on Low Realized Volatility): An investment strategy implementing long positions in low RV stocks and short positions in high RV stocks.

Kitchen Sink Model: A comprehensive model incorporating all available predictive variables for volatility forecasting.

Data availability

The high-frequency data are obtained from the Wind database (https://www.wind.com.cn). The other data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Andersen, T. G., & Bollerslev, T. (1998). Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *International Economic Review*, 885–905.

Andersen, T. G., Bollerslev, T., & Diebold, F. (2007). Roughing it up: Including jump components in the measurement, modelling and forecasting of return volatility. *The Review of Economics and Statistics, 89*, 701–720.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Ebens, H. (2001). The distribution of realized stock return volatility. *Journal of Financial Economics, 61*, 43–76.

Andersen, T. G., Bollerslev, T., & Meddahi, N. (2005). Correcting the errors: Volatility forecast evaluation using high-frequency data and realized volatilities. *Econometrica, 73*, 279–296.

Andersen, T. G., Dobrev, D., & Schaumburg, E. (2012). Jump-robust volatility estimation using nearest neighbor truncation. *Journal of Econometrics, 169*, 75–93.

Ang, A., Hodrick, R. J., Xing, Y., & Zhang, X. (2006). The cross-section of volatility and expected returns. *Journal of Finance, 61*, 259–299.

Atmaz, A. (2022). Stock return extrapolation, option prices, and variance risk premium. *The Review of Financial Studies, 35*, 1348–1393.

Audrino, F., & Knaus, S. D. (2016). Lassoing the HAR model: A model selection perspective on realized volatility dynamics. *Econometric Reviews, 35*, 1485–1521.

Audrino, F., Sigrist, F., & Ballinari, D. (2020). The impact of sentiment and attention measures on stock market volatility. *International Journal of Forecasting, 36*, 334–357.

Barndorff-Nielsen, O. E., & Shephard, N. (2004). *Measuring the impact of jumps in multivariate price processes using bipower covariation*. Discussion paper. Nuffield College, Oxford University.

Barndorff-Nielsen, O. E., & Shephard, N. (2006). Econometrics of testing for jumps in financial economics using bipower variation. *Journal of Financial Econometrics, 4*, 1–30.

BenSaida, A. (2021). The good and bad volatility: A new class of asymmetric heteroskedastic models. *Oxford Bulletin of Economics and Statistics, 83*, 540–570.

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

Bollerslev, T., Li, S. Z., & Zhao, B. (2020). Good volatility, bad volatility, and the cross section of stock returns. *Journal of Financial and Quantitative Analysis, 55*, 751–781.

Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5–32.

Bucci, A. (2020). Realized volatility forecasting with neural networks. *Journal of Financial Econometrics, 18*, 502–531.

Carr, P., Wu, L., & Zhang, Z. (2019). *Using machine learning to predict realized variance [preprint]*. arXiv. https://doi.org/10.48550/arXiv.1909.10035.

Chen, L., Pelger, M., & Zhu, J. (2024). Deep learning in asset pricing. *Management Science, 70*, 714–750.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference*. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2939672.2939785.

Chen, X. B., Gao, J., Li, D., & Silvapulle, P. (2018). Nonparametric estimation and forecasting for time-varying coefficient realized volatility models. *Journal of Business & Economic Statistics, 36*, 88–100.

Chen, Z., & Petkova, R. (2012). Does idiosyncratic volatility proxy for risk exposure? *Review of Financial Studies, 25*, 2745–2787.

Christensen, K., Siggaard, M., & Veliyev, B. (2023). A machine learning approach to volatility forecasting. *Journal of Financial Econometrics, 21*, 1680–1727.

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics, 7*, 174–196.

Cybenko, G. (1989). Approximations by superpositions of sigmoidal functions. *Mathematics of Control, Signals, and Systems, 2*, 303–314.

Dudek, G. (2015). Short-term load forecasting using random forests. In *, Vol. 2. Intelligent systems' 2014: Proceedings of the 7th IEEE international conference intelligent systems IS'2014* (pp. 821–828). Springer International Publishing.

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

Erel, I., Stern, L. H., Tan, C., & Weisbach, M. S. (2021). Selecting directors using machine learning. *The Review of Financial Studies, 34*, 3226–3264.

Fang, T., Lee, T. H., & Su, Z. (2020). Predicting the long-term stock market volatility: A GARCH-MIDAS model with variable selection. *Journal of Empirical Finance, 58*, 36–49.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters, 27*, 861–874.

Gong, X., & Lin, B. (2019). Modeling stock market volatility using new HAR-type models. *Physica A: Statistical Mechanics and its Applications, 516*, 194–211.

Gu, M., Kang, W., & Xu, B. (2018). Limits of arbitrage and idiosyncratic volatility: Evidence from China stock market. *Journal of Banking & Finance, 86*, 240–258.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies, 33*, 2223–2273.

Guo, H., & Savickas, R. (2010). Relation between time-series and cross-sectional effects of idiosyncratic variance on stock returns. *Journal of Banking & Finance, 34*, 1637–1649.

Guo, H., Tang, S., & Wei, Z. (2024). The information role of Investors' site visits in management forecasts. *Emerging Markets Finance and Trade, 60*(8), 1762–1794.

Izzeldin, M., Hassan, M. K., Pappas, V., & Tsionas, M. (2019). Forecasting realised volatility using ARFIMA and HAR models. *Quantitative Finance, 19*, 1627–1638.

Izzeldin, M., Muradoğlu, Y. G., Pappas, V., & Sivaprasad, S. (2021). The impact of Covid-19 on G7 stock markets volatility: Evidence from a ST-HAR model. *International Review of Financial Analysis, 74*, Article 101671.

Jiang, G. J., & Oomen, R. C. (2008). Testing for jumps when asset prices are observed with noise–a "swap variance" approach. *Journal of Econometrics, 144*, 352–370.

Jiang, J., Kelly, B., & Xiu, D. (2023). (re-) Imag (in) ing price trends. *The Journal of Finance, 78*, 3193–3249.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems, 30*.

Koopman, S. J., Jungbacker, B., & Hol, E. (2005). Forecasting daily variability of the S&P 100 stock index using historical, realised and implied volatility measurements. *Journal of Empirical Finance, 12*, 445–475.

Leippold, M., Wang, Q., & Zhou, W. (2022). Machine learning in the Chinese stock market. *Journal of Financial Economics, 145*, 64–82.

Li, K., & Liu, J. (2023). Extrapolative asset pricing. *Journal of Economic Theory, 210*, Article 105651.

Liu, J., Ma, F., Yang, K., & Zhang, Y. (2018). Forecasting the oil futures price volatility: Large jumps and small jumps. *Energy Economics, 72*, 321–330.

Lochstoer, L. A., & Muir, T. (2022). Volatility expectations and returns. *The Journal of Finance, 77*, 1055–1096.

Luo, J., Klein, T., Ji, Q., & Hou, C. (2022). Forecasting realized volatility of agricultural commodity futures with infinite hidden Markov HAR models. *International Journal of Forecasting, 38*, 51–73.

Luo, Q., Ma, F., Wang, J., & Wu, Y. (2024). Changing determinant driver and oil volatility forecasting: A comprehensive analysis. *Energy Economics, 129*, Article 107187.

Masters, T. (1993). *Practical neural network recipes in C++*. Morgan Kaufmann.

Motegi, K., Cai, X., Hamori, S., & Xu, H. (2020). Moving average threshold heterogeneous autoregressive (MAT-HAR) models. *Journal of Forecasting, 39*, 1035–1042.

Patton, A. J., & Sheppard, K. (2015). Good volatility, bad volatility: Signed jumps and the persistence of volatility. *Review of Economics and Statistics, 97*, 683–697.

Rahimikia, E., & Poon, S. H. (2020). *Machine learning for realized volatility forecasting.* Available at SSRN, 3707796.

Song, Y., Lei, B., Tang, X., & Li, C. (2024). Volatility forecasting for stock market index based on complex network and hybrid deep learning model. *Journal of Forecasting, 43*, 544–566.

Su, Z., Cai, X., & Wu, Y. (2023). Exchange rates forecasting and trend analysis after the COVID-19 outbreak: New evidence from interpretable machine learning. *Applied Economics Letters, 30*(15), 2052–2059.

Vrontos, S. D., Galakis, J., & Vrontos, I. D. (2021). Implied volatility directional forecasting: A machine learning approach. *Quantitative Finance, 21*, 1687–1706.

Wang, L., Ma, F., Liu, J., & Yang, L. (2020). Forecasting stock price volatility: New evidence from the GARCH-MIDAS model. *International Journal of Forecasting, 36*, 684–694.

Wang, Y., Ma, F., Wei, Y., & Wu, C. (2016). Forecasting realized volatility in a changing world: A dynamic model averaging approach. *Journal of Banking & Finance, 64*, 136–149.

Wen, F., Zhao, Y., Zhang, M., & Hu, C. (2019). Forecasting realized volatility of crude oil futures with equity market uncertainty. *Applied Economics, 51*, 6411–6427.

Wilms, I., Rombouts, J., & Croux, C. (2021). Multivariate volatility forecasts for stock market indices. *International Journal of Forecasting, 37*, 484–499.

Xiao, J., Wen, F., Zhao, Y., & Wang, X. (2021). The role of US implied volatility index in forecasting Chinese stock market volatility: Evidence from HAR models. *International Review of Economics & Finance, 74*, 311–333.

Xie, H., Sun, Y., & Fan, P. (2023). Return direction forecasting: A conditional autoregressive shape model with beta density. *Financial Innovation, 9*(1), 82.

Yang, R., Yu, L., Zhao, Y., Yu, H., Xu, G., Wu, Y., & Liu, Z. (2020). Big data analytics for financial market volatility forecast based on support vector machine. *International Journal of Information Management, 50*, 452–462.

Yun, S., Oh, S. J., Heo, B., Han, D., Choe, J., & Chun, S. (2021). Re-labeling ImageNet: From single to multi-labels, from global to localized labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 2340–2350).

Zhang, C., Zhang, Y., Cucuringu, M., & Qian, Z. (2023). Volatility forecasting with machine learning and intraday commonality. *Journal of Financial Economics, 22*, 492–530.

Zhang, Z., & Zhao, R. (2023). Good volatility, bad volatility, and the cross section of cryptocurrency returns. *International Review of Financial Analysis, 89*, Article 102712.