ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 2 - 2026

Forecasting financial market volatility and its risk assessment in digital economy based on time series computational analysis

Qi Deng[1,*]

[1] Business School, Zhengzhou Technology and Business University, Zhengzhou, 450000, Henan, P.R.China

SUMMARY: *This paper centers on the problem of predicting the volatility of financial markets and assessing their risks. Firstly, a financial market forecasting method based on EEMD-LSTM is realized. The realized volatility forecasting model based on EEMD-LSTM is constructed with the realized volatility series of CSI 300 index as the research object. And RMSPE, R^2 and MAE are used as the evaluation indexes of the model's forecasting effect to compare the experiments with the comparison model. Then the VaR calculation method based on Copula function to improve the traditional Monte Carlo simulation (MC) method is proposed for financial market risk assessment. Applied to the trading data of PetroChina stock market, the calculation results show that the Copula-MC method is more accurate and effective in approaching the actual VaR.When the EEMD-LSTM-based financial market prediction method and the risk assessment method based on Copula-MC are used in combination, the effect of estimating VaR is more satisfactory, and the method in this paper can effectively predict the volatility of the financial market and compute the value of its risk.*

KEYWORDS: *EEMD-LSTM; Copula-MC; VaR; financial market*

1 Introduction

Market volatility is one of the key indicators of financial analysis, which is used to assess the future risk, the magnitude of asset price fluctuations and reflect the uncertainty of asset returns [1]. Volatility is widely used in financial investment and risk management, and is an important reference indicator for investors' decision-making and risk control [2]. At the same time, volatility has an important impact on prices and is a key basis for constructing quantitative investment strategies for options [3, 4]. Therefore, analyzing and predicting the market volatility of financial data is of great significance for financial investors.

Currently, the financial trading market mainly uses traditional financial volatility models [5]. Examples include ARMA (Auto-Regressive Moving Averagel) model [6], GARCH (Generalized Auto-Ressive Conditional Heteroskedasticity) model [7] and SV (Stochastic Volatikity) models [8] to forecast financial market volatility. It is based on the relationship between current and lagged volatility and uses the conditional variance formula and historical volatility to derive the behavioral characteristics of the current volatility and to forecast it. Literature [9] constructed a model for predicting the volatility of stock index returns by combining asymmetric GARCH model with implied volatility through ARMA model. Literature [10] proposed a hybrid forecasting model combining wavelet transform, ARIMA and

*13673389286@163.com
https://doi.org/10.65102/is2026980
Published: 30 April, 2026

GARCH models for improving volatility forecasting of Tesla stock. Literature [11] compared traditional combination models such as ARMA and GARCH to predict the statistical performance and out-of-sample prediction accuracy of the S&P 500 index for the period 2006-2010, which is not satisfactory for its stock market volatility. Combining the above literature, it is found that these traditional financial models can only deal with linear relationships in the time series and have limited ability to deal with nonlinear relationships in the time series [12]. In addition, these methods require manual processing for multivariate synergistic prediction and are mainly applicable to small-scale univariate time series prediction problems [13].

With the development of the digital economy and the complexity of financial markets, more and more researchers have begun to explore the application of deep learning in volatility prediction in financial markets [14]. Literature [15] attempted to improve the traditional GARCH prediction model using SVM (Support Vector Machine) model and outperformed the traditional method with better generalization performance and higher hit rate on simulated and real datasets. Literature [16] proposed a deep learning algorithm based on LSTM (Long Short-Term Memory Network) to accurately predict stock market indices and their volatility and evaluated the performance on 7 years of data for 5 stock market indices with better results than other models. Literature [17] proposed a new hybrid model that combines the traditional GARCH model with the distribution manipulation strategy of LSTM, aiming to improve the prediction performance of stock market volatility. Literature [18] proposes a spatio-temporal GNN (Convolutional Neural Network) overflow model that outperforms the benchmark model in short- and medium-term financial market volatility prediction, and investors can gain economic benefits from volatility prediction of this model. The above deep learning methods for financial volatility prediction are difficult to effectively capture key time series features due to the nonlinearity and long-term dependence of time series data [19]. In addition, due to the problem of vanishing gradient, these its models cannot fully utilize the past information for forecasting, which leads to a decrease in the predictive ability of the models [20]. Time-series computational analysis can solve the problems that traditional deep learning models are ineffective in capturing nonlinearities and long-term dependencies of time-series data, as well as being prone to gradient vanishing [21].

In this paper, we propose a financial market forecasting method based on EEMD-LSTM, firstly, the empirical modal decomposition method EEMD is used to decompose the time series, the original financial market series are processed, and the variables are obtained after decomposition. Then the components are input into the LSTM model to forecast the financial market, and the results of the predicted components are summed to get the final forecast results. The financial risk assessment method uses Copula function to improve the traditional Monte Carlo simulation method to calculate VaR.The study adds the EEMD-LSTM combination model to the Mento Carlo simulation method, and uses the volatility estimated by the EEMD-LSTM combination model to calculate the VaR value, so as to judge the effect of risk assessment.

2 Research on forecasting financial market volatility

2.1 Predictive modeling

This chapter combines the empirical modal decomposition method and the long- and short-term memory network to construct an EEMD-LSTM model to realize the volatility prediction of financial markets.

2.1.1 Long and short-term memory networks

LSTM is a deep learning model for modeling sequence data and is a novel neural network architecture that effectively solves problems such as gradient vanishing and gradient explosion in RNNs, enabling the model to handle complex long sequence data, thus improving the accuracy and reliability of the model [22]. Each unit in LSTM contains a gating mechanism that automatically controls the input of information, output and forgetting, thus enabling the modeling of long term dependencies in sequence data.

The core idea of LSTM is to take the input of the current time step and the hidden state of the previous time step as inputs, and then control the flow and forgetting of information through the gating mechanism.LSTM networks have been widely used in many sequence learning tasks, and the architecture of LSTM networks is shown in Fig. 1. Specifically, LSTM contains three gating units: input gate, forgetting gate and output gate.

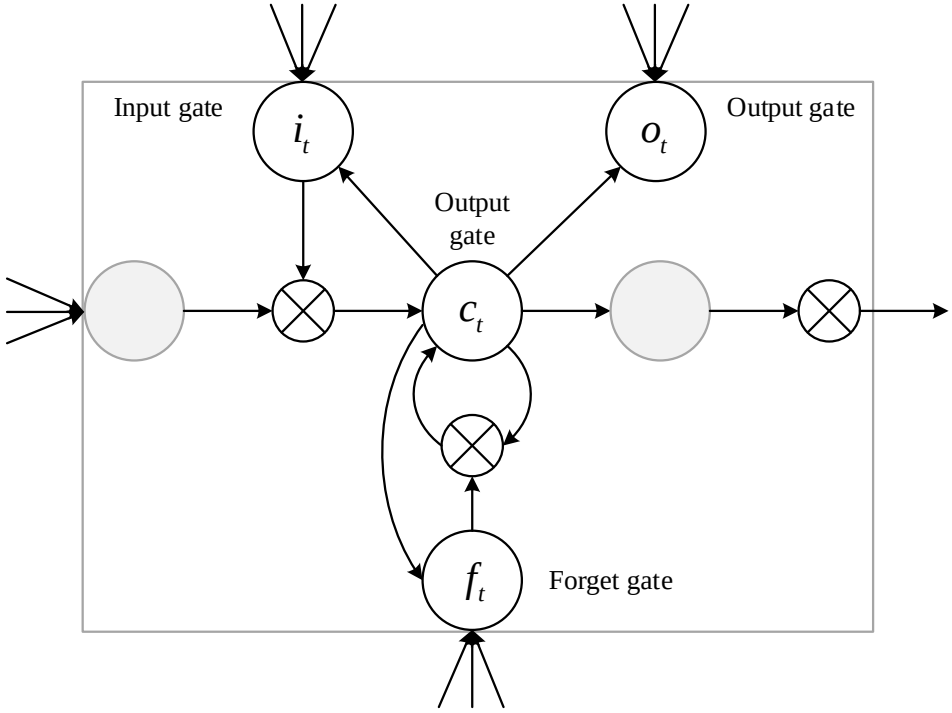


Figure 1: LSTM Network Architecture

The LSTM network structure has three gates:

Input gate: the input gate is used to calculate which information is saved into the state unit, including two parts of information, one part:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{1}$$

This section can be viewed as how much information needs to be saved to the unit state for the current input. The other part is:

$$\tilde{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{2}$$

This part is used to add new information generated by the current input to the unit state. These two parts produce a new memory state and the unit state at the current moment consists

of the product of the forget gate input and the state at the previous moment plus the product of the two parts of the input gate, i.e.:

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \tag{3}$$

Oblivion Gate: The oblivion gate is used to calculate the extent to which information is forgotten (retained), and is a value from 0 to 1 after sigmoid processing, with 1 indicating that it is all retained and 0 indicating that it is all forgotten.

Output Gate: Used to calculate the extent to which information is output at the current moment:

$$o_t = \sigma(W_0 \cdot [h_{t-1}, x_t] + b_0) \tag{4}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{5}$$

2.1.2 Collective Empirical Modal Decomposition

Empirical modal decomposition (EMD) can effectively extract complex signal features for better understanding and prediction of signals.The EMD method has an extremely strong instantaneous frequency decomposition capability, and thus can be applied to a variety of complex signal decomposition problems, especially for nonsmooth, nonlinear time series [23].

Although the empirical modal decomposition technique can effectively deal with nonsmooth, nonlinear signals, due to the complexity of the nature of the signals and the limitations of the algorithms, the ultimately obtained IMFs may suffer from modal aliasing, i.e., the appearance of different eigencomponents at different time scales. Since the decomposition process of IMFs requires several iterations, but the current lack of a unified reference standard makes the results of each iteration potentially different. Therefore, the EEMD technique can effectively overcome the modal confusion problem in the EMD method and thus provide more accurate results.The EEMD method can effectively overcome the limitations of the EMD method by introducing multiple white noises at different scales and the zero-mean feature, which can obtain the IMFs from multiple time scales, and then, by taking the mean value, the differences between these IMFs are effectively reduced, thus obtaining more accurate analysis results.

Decomposition steps:

(1) Add white noise to the original signal:

$$I_i(t) = I(t) + \alpha_i(t) \tag{6}$$

(2) EMD decomposition of $I_i(t)$ obtained by performing the noise addition process to obtain IMFs:

$$I_i(t) = \sum_{j=1}^{n} imf_{ij}(t) + r_{in}(t) \tag{7}$$

i represents the i th decomposition, and j represents the j th result obtained from each decomposition.

(3) Repeat the above process by adding different white noise several times to change the original signal and decompose it using EMD.

(4) Mean the results obtained after m times of repeated decomposition, which is done by averaging the imf and trend terms of the same order to obtain the final imf and trend terms respectively:

$$imf_j(t) = \frac{1}{m}\sum_{i=1}^{m} imf_{ij}(t) \tag{8}$$

$$r_n(t) = \frac{1}{m}\sum_{i=1}^{m} r_{in}(t) \tag{9}$$

(5) The final result of EEMD decomposition of the original signal is obtained:

$$I(t) = \sum_{j=1}^{n} imf_j(t) + r_n(t) \tag{10}$$

According to the above EEMD decomposition process, the number of trials for adding noise is set to 300, and the width of the added Gaussian white noise is set to 0.01. In this paper, EEMD decomposition of the original data is performed using Matlap.

2.1.3 EEMD-LSTM modeling

The EEMD-LSTM model proposed in this paper [24] is divided into two parts, and the specific flowchart is shown in Fig. 2.

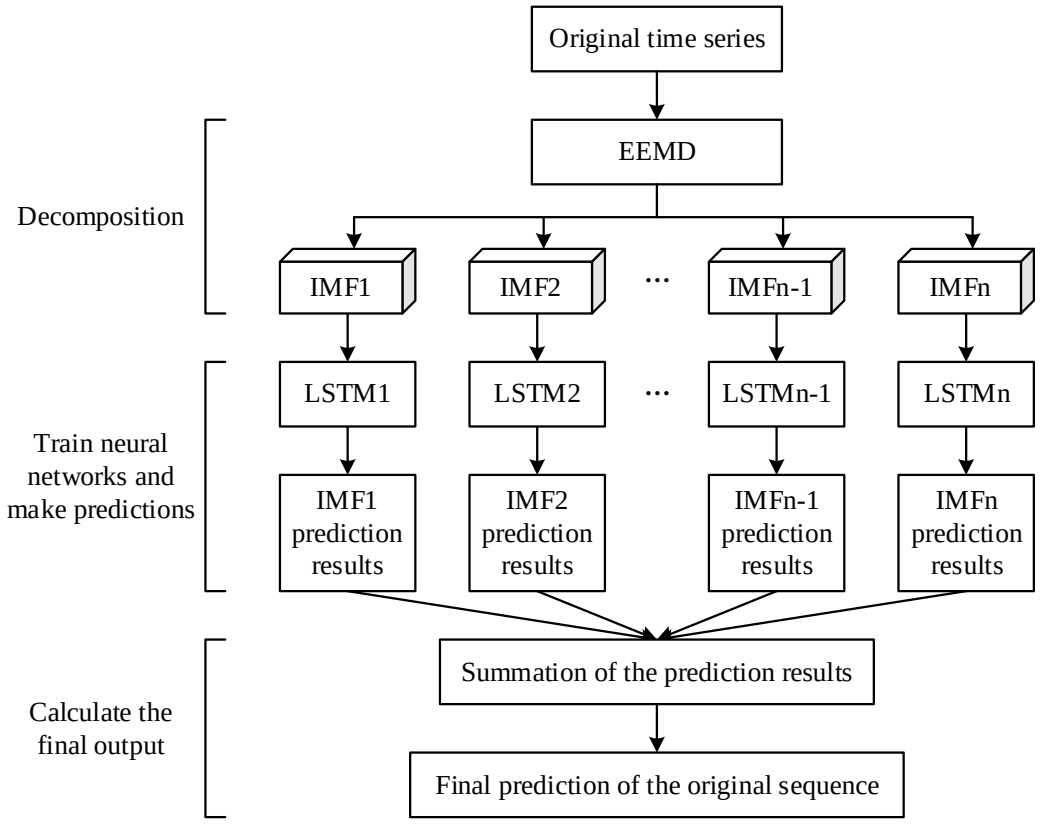


Figure 2: The overall process of the EEMD-LSTM prediction model

The first part is based on the EEMD to realize the financial market time series

decomposition, which can find out the inner pattern and decompose into multiple IMF components without too much human settings and interventions.

The second part is based on LSTM model to realize financial market forecasting. The IMF components after EEMD decomposition are inputted into the constructed financial market LSTM prediction model, and IMF1-7 are used as the training set, IMF8-9 as the testing set, and IMF10 as the validation set. After the model training is completed, the financial market data IMF8-9 from the test set are fed into the prediction model to obtain the prediction results, and the prediction results of each component are summed up to obtain the final predicted price. Finally, by comparing the predicted price with the actual price, we can evaluate the accuracy of the model to better grasp the prediction performance of the model.

2.2 Experimental Simulation and Analysis

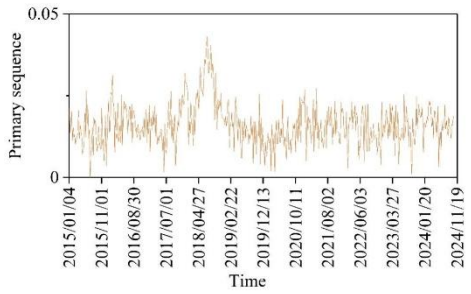
2.2.1 Analysis of volatility forecasts

(1) Sample Selection

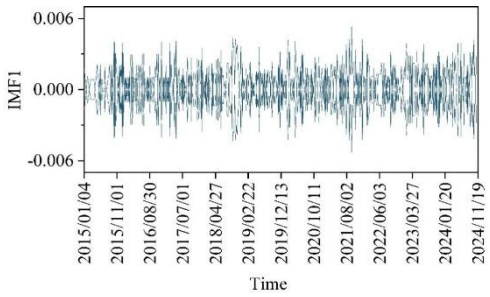
In this paper, the sample data of CSI 300 index is selected as the research object to forecast the realized volatility, with the time span from January 4, 2015 to December 30, 2024. Since the calculation of realized volatility requires the use of intraday high-frequency data, and the acquisition of high-frequency data is sensitive to the sampling frequency, this paper selects a 5-minute sampling frequency that can better balance the accuracy of the data and the microstructural noise, in which the relevant data are obtained from the financial data API of the Polybroad Quantitative Platform.The characteristics of the selected data are mainly related to the price and trading volume of the CSI 300 index, mainly the price of the CSI 300 index and the trading volume of the CSI 300 index. The main types of data are volume, turnover, minimum price, maximum price, opening price and closing price of CSI 300 index.

(2) Decomposition and reconstruction of realized volatility

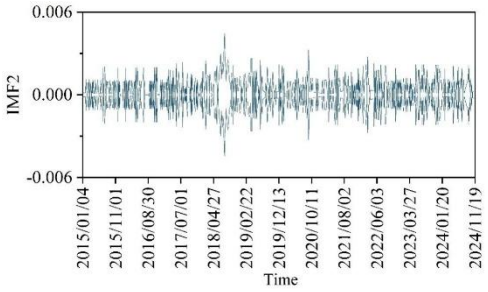
The essence of the ensemble empirical modal method is to continuously decompose the nonlinear time series, decompose the original seemingly irregular time series into regular eigenmodal function components, and then analyze the eigenmodal function components to further study the characteristics of the realized volatility, which will help us accurately carry out the subsequent model prediction. Through the principle of ensemble empirical modal decomposition described in the previous theory, this paper decomposes the daily realized volatility series of CSI 300 index which has been calculated, and the final decomposition results of realized volatility are shown in Fig. 3. Figure (a) shows the original sequence of CSI 300 index from January 4, 2015 to December 30, 2024, figure (k) shows the residual term, and figure (b~j) shows the sequence of intrinsic modal functions IMF1~9 obtained by decomposition.From its decomposition results, it can be clearly seen that from IMF1 to IMF9, the oscillation frequency of each intrinsic modal function component becomes lower and lower, and after the oscillation frequency decreases to a certain degree, the trend of the eigenmode function components becomes very flat, and finally the curve represented by the residual term shows monotonicity.



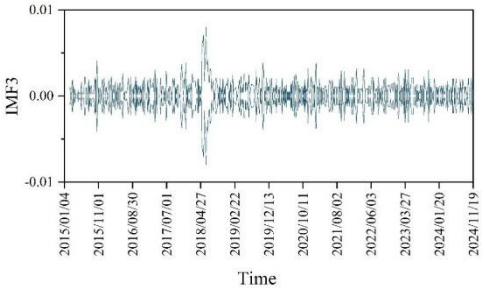
(a) Primary sequence



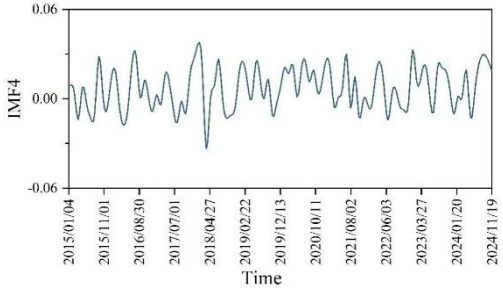
(b) IMF1



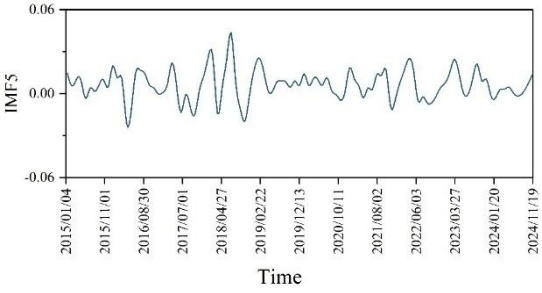
(c) IMF2



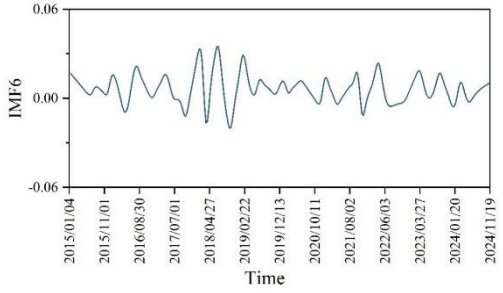
(d) IMF3



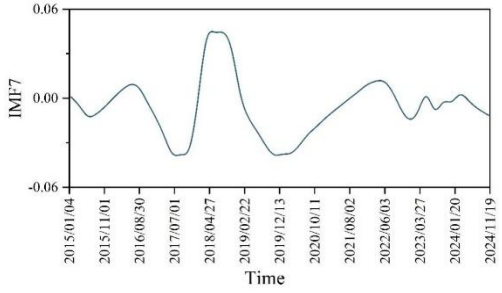
(e) IMF4



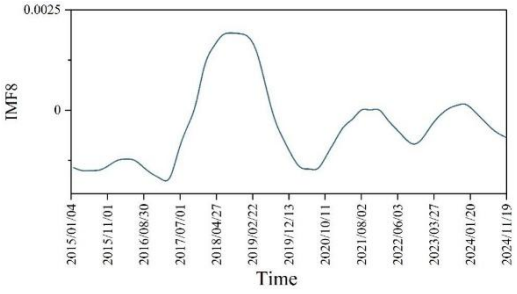
(f) IMF5



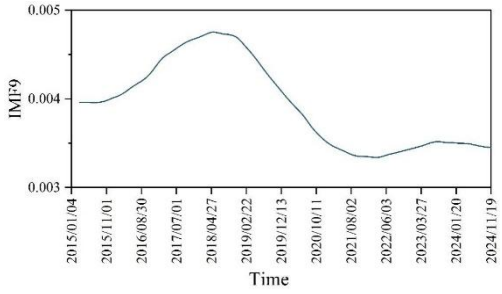
(g) IMF6



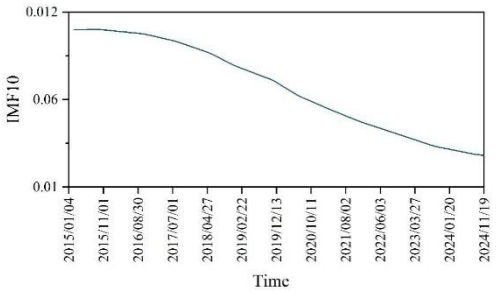
(h) IMF7



(i) IMF8



(j) IMF9



(k) Residual term

Figure 3: The collection experience modal decomposition has been realized

Next, this paper performs descriptive statistics on each component obtained from the decomposition to compare the relationships implied between the sequences of components.

In Fig. 3, we can find that the eigenmode function components with higher oscillation frequencies are fluctuating up and down around the 0-axis. The amplitude of the oscillations is clearly larger for the eigenmode function components with lower oscillation frequencies. In order to verify this law, the descriptive statistics of each intrinsic modal function component and the residual term after the decomposition are counted, and the statistical results are shown in Table 1. From the mean value of each component, it seems that the mean value of each component is close to 0. In order to further test the difference between the high oscillation frequency component and the low oscillation frequency component, this paper has conducted a hypothesis test on the significance of the mean value of 0 for each component at a significance level of 5%, and the P-value in the last column of Table 1 is the result of the hypothesis test on whether the mean value of the sequence of each component is 0 or not. As can be seen from the results in this column, IMF1 to IMF6 accepted the 0-mean significance hypothesis test, indicating that the means of these eigenmode function components are significantly 0, while IMF7 to IMF9 and the residual term (IMF10) rejected the 0-mean significance hypothesis test, indicating that the means of these components are not significantly 0.

Table 1: Descriptive statistics of each component

	Mean	Median	SD	P
IMF1	-3.6E-5	-8.7E-5	0.002155	0.2665
IMF2	-8E-6	-1.8E-5	0.001164	0.4159
IMF3	-3E-5	-1.8E-5	0.00155	0.5001
IMF4	1.3E-5	-3.5E-5	0.00136	0.7051
IMF5	-3.7E-5	-5.6E-5	0.001377	0.0996
IMF6	-1.7E-5	-1.09E-4	0.001367	0.5065
IMF7	-3.46E-4	-2.57E-4	0.002422	0.0218
IMF8	6.2E-5	-1.51E-4	0.001554	0.0397
IMF9	0.00388	0.0039	0.000845	0
IMF10	0.01039	0.01045	0.001194	0

In the previous section we have derived that the mean value of the intrinsic modal function components from IMF1 to IMF6 is significantly 0, the mean value of the components from IMF7 to IMF9 is not significantly 0, and the oscillation frequency of each intrinsic modal function component is very high in IMF1 to IMF6. In IMF7 to IMF9, the oscillation frequency of each eigenmode function component is relatively low. Thus, in this paper, IMF1 to IMF6 are reconstructed as a high vibration frequency sequence (abbreviated as high frequency sequence)

and IMF7 to IMF9 are reconstructed as a low vibration frequency sequence (abbreviated as low frequency sequence) based on the common feature of oscillation frequency of the eigenmodal function, and the residual term is analyzed separately as a trend term.

Next, the three components of the reconstructed high-frequency sequence, low-frequency sequence, and residual term are analyzed by descriptive statistics and plotted as corresponding trend terms, and the results are shown in Table 2 and Fig. 4. The low-frequency series is equivalent to a smoothing treatment of the original realized volatility series, and the low-frequency series presents a certain regularity.

In summary, the use of the ensemble empirical modal decomposition algorithm can decompose and reconstruct the regular eigenmode function components from the seemingly irregular realized volatility series.

Table 2: Descriptive statistics of each component of the decompostion reconstruction

	Mean	Max	Min	Median	SD	Kurtosis	Degree of bias
High frequency sequence	0.0001	0.0407	-0.0132	-0.0012	0.0049	20.4386	3.1013
Low frequency sequence	0.0039	0.0151	-0.001	0.0048	0.0032	2.59546	1.6634
Residual term	0.0108	0.0097	0.0079	0.0112	0.0016	-1.4854	-0.0833

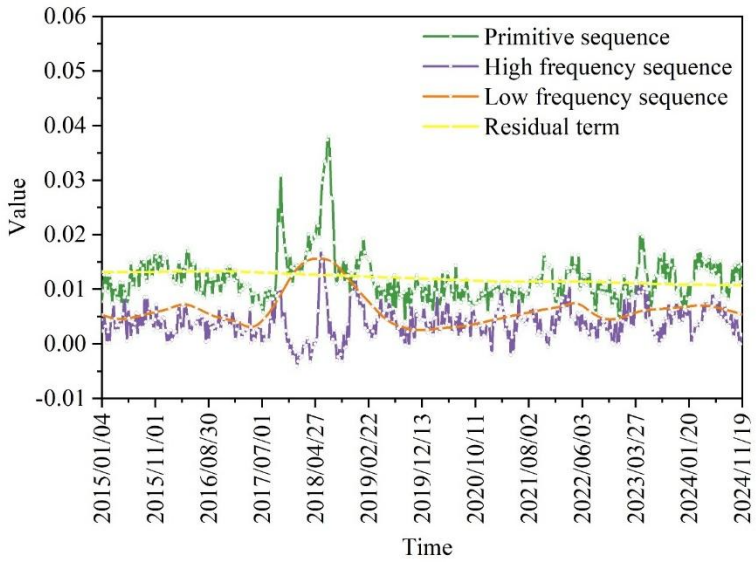


Figure 4: The EEMD decomposition and the original sequence trend

2.2.2 Prediction Error Comparison Experiment

(1) Experimental assessment metrics

The experiment uses the mean squared percentage error (RMSPE), R^2 and MAE, which are commonly used in the three regression problems, as the assessment indexes of the model prediction effect. As the value of short-term volatility of financial high-frequency trading data is on the small side, RMSPE is able to eliminate the influence brought by its data size, and can reflect the overall level of prediction results. RMSPE is calculated as the percentage error for each data point to seek the ordinary and average and then take the square root of the percentage error, and the specific calculation method is as shown in Equation (11):

$$RMSPE = \frac{\sqrt{\sum_{i=1}^{n}[\frac{y_i - y_i^{'}}{y_i}]^2}}{n} \tag{11}$$

The R^2 indicator is used in the regression task to assess the fit of the model to the data, it indicates the proportion of the variance predicted by the model to the actual variance, the value ranges from 0-1, the closer it is to 1 means that the model fits the data better, the R^2 is calculated as 1 minus the ratio of the sum of squares of the residuals to the sum of the total squares, and the specific calculation method is shown in Equation (12):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - y_i^{'})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{12}$$

The MAE indicator is to calculate the difference between the predicted value and the real value, the smaller the value of MAE, the smaller the prediction error of the model, the better the prediction of the model.The calculation method of MAE is shown in equation (13):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}| y_i - y_i^{'} | \tag{13}$$

In the above formula, y_i denotes the actual value of the volatility of the financial high frequency trading data, $y_i^{'}$ denotes the predicted value of the model on the volatility of the financial high frequency trading data, n denotes the number of samples in the data set, and $\bar{y}$ denotes the average value of the volatility of the financial high frequency trading data.

(2) Experimental Data

The data used for the experiments in this chapter are real-time high-frequency trading data of the 50ETF from September 2023 to April 2024, obtained in real-time from Wind's third-party trading platform. The dataset includes both order book data and trading data. In this paper, the daily market order book data and trading data are divided into 10-minute time windows to predict the volatility of the 50ETF fund in the next 10 minutes.

(3) Experimental Results

In order to compare and analyze the prediction effect of the EEMD-LSTM model proposed in this chapter, the prediction effect of the model is compared with that of other benchmark models. The models used for comparison in this paper include traditional machine learning models, commonly used time series forecasting models and unaltered ns Transformer, which include machine learning models LGBM, Random Forest, and commonly used deep learning models LSTM, NN, CNN for time series forecasting.The results are computed by using R^2, RMSPE, and MAE for different forecasting methods in the 50ETF high-frequency trading dataset, and the results are shown in Table 3. The traditional machine learning algorithms are less effective on the test set, and after using wavelet smoothing on the data, the effect decreases instead, the deep learning model combined with DAE effect has a significant improvement, and the prediction effect of the deep learning model is not as good as that of the EEMD-LSTM model proposed in this paper.The EEMD-LSTM model has the best prediction and fitting effect among all the models. Compared with the LSTM model, R^2 improves from 0.8615 to 0.9084, RMSPE decreases from 0.2267 to 0.1554, and MAE decreases from 5.696E-4 to 3.317E-4. The experimental results prove that the accuracy of the prediction of the model proposed in this paper is greatly improved compared with other models.

Table 3: Comparison experiments of different models

Model	R^2	RMSPE	MAE
KFord_LGBM	0.8343	0.2389	5.76E-4
KFord_LGBM_NN	0.869	0.243	5.567E-4
Wavelet_LGBM_NN	0.7491	0.3042	7.784E-4
Random Forest	0.7782	0.3079	6.268E-4
LSTM	0.8615	0.2267	5.696E-4
CNN	0.8632	0.2228	5.363E-4
DAE_LSTM	0.8904	0.1982	4.45E-4
DAE_CNN	0.8933	0.1886	4.491E-4
ns_Transformer	0.8626	0.2251	4.736E-4
EEMD-LSTM	0.9084	0.1554	3.317E-4

3 Financial market risk assessment

3.1 Optimization of the assessment methodology

VaR technique is an important method in risk management, and Monte Carlo simulation (MC) method to calculate VaR has been widely used in practice [25], but it relies too much on assumed good distributions and models in the generation of pseudo-random numbers and the determination of joint distributions. In this chapter, Copula function is used to improve the traditional MC method [26] to realize the effective assessment of financial market risk.

3.1.1 Traditional MC methods

The basic idea of VaR calculation based on MonteCarlo simulation method is to repeat the simulation of stochastic processes of financial variables so that the simulated values include most of the possible scenarios, so that the overall distribution of portfolio values can be obtained through simulation to find VaR. it is divided into the following four main steps:

In the first step, the confidence level required for VaR computation is chosen $1-\alpha$.

In the second step, a pseudo n pseudo-random sequence is generated under an appropriate joint distribution describing the risk factors and the price sequence $V_{t+1,1}, V_{t+1,2}, \cdots, V_{t+1,m}$ is computed.

In the third step, simulated gains and losses are computed under this price series $DeltaV_I = V_{t+1,i} - V_t, (i = 1, 2, \cdots, m)$.

In the fourth step, the worst ΔV_i under the α quartile is ignored, and the smallest value of the remaining ΔV_i is the VaR at time t, defined as VaR $(\alpha, t, t+1)$. When time goes from t to $t+1$, the price series changes from V_t to V_{t+1}, and we can return to test the VaR $(\alpha, t, t+1)$ by comparing ΔV, and repeating until the simulation requirements are met.

Obviously there are two main steps in the MC method, one of which is the generation of pseudo-random numbers and the other is the determination of the joint distribution, which cannot be handled well by traditional methods, and the following discussion centers around these two issues by Copula means.

3.1.2 Copula MC method

In this paper, we first give the traditional algorithm for generating pseudo-random numbers and improve the steps in it using Copula method. If the foreign exchange exchange rate as a risk

factor to be calculated, the traditional method of generating pseudo-random numbers includes the following steps:

The first step collects n exchange rate historical data with a time series spanning N+1 days, denoted as $x_{i,0}, x_{i,1}, \cdots, x_{i,N} (i = 1, \cdots, n)$, and the current day is $x_{i,N}$, which is generally chosen as N+1=250 or 500.

The second step assumes $x_{i,j} \neq 0$, and calculates the relevant changes from the data:

$$r_{i,j} = \frac{x_{i,j} - x_{i,j-1}}{x_{i,j-1}}, i = 1, \cdots n; \; j = 1, \cdots, N, r_{i,j} \in \{r\} \tag{14}$$

The third step assumes that the marginal distribution of the random variable $r_1, \ldots, r_n$ is $f_1, \ldots, f_n$, and calculates the corresponding parameters. In exchange rate risk calculations, a normal distribution $N(\mu_i, \sigma_i^2)$ is usually assumed, i.e.:

$$f_i(r_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left\{ -\frac{(r_i - \mu_i)^2}{2\sigma_i^2} \right\}, \hat{\mu}_i = \frac{1}{N} \sum_{j=1}^{N} r_{i,j}, \hat{\sigma}_i^{\,2} = \frac{1}{N-1} \sum_{j=1}^{N} (r_{i,j} - \hat{\mu}_i)^2 \tag{15}$$

In the fourth step, the multivariate joint distribution is assumed to be:

$$f(r = \frac{1}{\sqrt{(2\pi)^n \det C}} \exp\left\{ -\frac{1}{2} (r - u^{-1} C^{-1} (r - u) \right\} \tag{16}$$

Among them:

$$r = \begin{pmatrix} r_1 \\ \vdots \\ r_n \end{pmatrix}; \mu = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix}; C = \begin{pmatrix} \sigma_1^2 & c_{1,2} & \cdots & c_{1,n} \\ c_{1,2} & \sigma_2^2 & \cdots & c_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1,n} & c_{2,n} & \cdots & \sigma_n^2 \end{pmatrix}; c_{i,j} = E((r_i - \mu_i)(r_j - \mu_j)) \tag{17}$$

The fifth step calculates the covariance array:

$$\hat{c}_{i,j} = \frac{1}{N-1} \sum_{j=1}^{N} (r_{i,k} - \hat{\mu}_i)(r_{j,k} - \hat{\mu}_j) \tag{18}$$

Step 6 generates pseudo-random numbers. Firstly, Cholesky decomposition of C, $C^{-1} = A^T A$, A is a lower triangular array that generates independent random variables $s_1, s_2, \ldots, s_n$ on [0, 1]. Then a sequence of pseudo-random numbers $r_1, r_2, \ldots, r_n$ is obtained according to $r = A^{-1} s + \mu$, and this is repeated to obtain $r^k = (r_1^k, \cdots, r_n^k)^T$, $k = 1, \cdots, m$ is the number of MC simulations.

To address the shortcomings of the above algorithm, the following is an improvement with the Copula method. The first three steps are kept consistent with the original method and the Copula method is introduced in the fourth, fifth and sixth steps.

In the fourth step, for the joint distribution function of two risk factors:

$$C_\theta(\varphi_1(r_1),\varphi_2(r_2)) = P(R_1 \le r_1, R_2 \le r_2) \tag{19}$$

where $\varphi_i(r_i) = \int_{-\infty}^{r_j} dr_i' f_i(r_i'$ is the cumulative density function.

In the fifth step, θ is estimated by the great likelihood method. Let $f_\theta(r_1,r_2) = \frac{\partial^2}{\partial r_1 \partial r_2} C_\theta(\varphi_1(r_1),\varphi_2(r_2))$, the likelihood function is:

$$L(\theta) = \prod_{j=1}^{N} f_\theta(r_{1,j}, r_{2,j}) \tag{20}$$

An estimate of $l(\theta) = \ln L(\theta)$ can be obtained as $I(\theta)\sum_{j=1}^{n} \ln\left\{\frac{\partial^2}{\partial u \partial v} C_\theta(u,v)\,|_{u=\varphi_1(r_{1,j}),v=\varphi_2(r_{2,j})}\right\}$, which in turn can be solved for $\hat{\theta}$

In the sixth step, generate two independently normally distributed pseudo-random numbers u, w on [0, 1] and compute $v = C_{\hat{\theta},u}^{-1}(w)$, where $C_{\hat{\theta},u} = \frac{\partial}{\partial u} C_{\hat{\theta}}(uv)$, such that $r_1 = \varphi_1^{-1}(u), r_2 = \varphi_2^{-1}(v)$ This gives the pseudo-random number pair (r_1, r_2).

3.2 Calculation of VaR based on the improved MC simulation method

The above improvement of MC simulation method is just a simple analysis based on the theory, and not a mathematical analysis of its statistical characteristics. In this section, we apply statistical methods to mathematically analyze the stock data to verify the sharp peaks and thick tails characteristics of the market returns and the volatility aggregation phenomenon.

We conduct a statistical test on the number of PetroChina stocks. It is known that its interval is from 1 & 1, 2019 to December 31, 2024, with a total of 1,200 trading days of closing prices. Firstly, its daily return is calculated with the following formula:

$$R_t = \log P_t - \log P_{t-1} \tag{21}$$

where P_t denotes the closing price of the stock on the t th trading day. A total of 1199 market index returns are obtained. The normality and volatility aggregation are tested below, and the software used is Eviews.

3.2.1 Normality test

Experimental Q-Q plots and Jarque-Bara method were tested.The results of the Q-Q plot test are shown in Figure 5. According to the Q-Q plot normality criterion, when the return distribution is normal, the Q-Q plot is a straight line. The Q-Q plot lines of this data set are all S-shaped, so it can be preliminarily judged that the distribution of the stock's return is not normal.

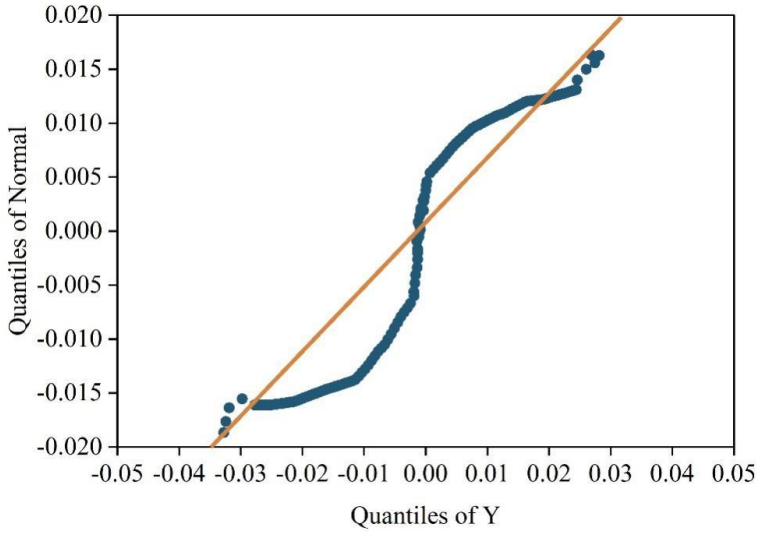


Figure 5: Chinese oil stock yield of Q-Q diagram

The results of the Jarque-Bara test for this data set are shown in Figure 6. The critical value of the JB statistic at the 95% level of significance is 5.99. As seen in Figure 6, the JB statistic for this data set is 7517.691, all of which are much larger than the critical value of 5.99, indicating that the returns do not follow a normal distribution.

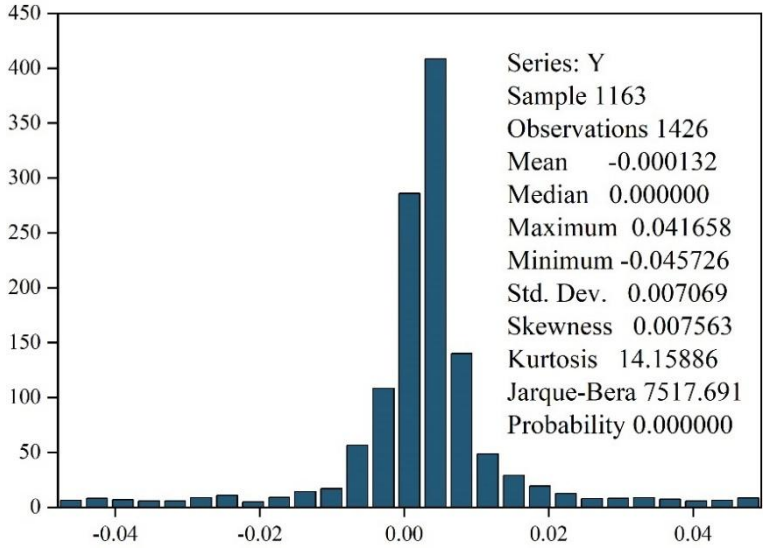


Figure 6: Chinese oil stock yield of Jarque-Bara test

3.2.2 Volatility Aggregation Test

First of all, it is necessary to have an intuitive understanding of the volatility of the pairs of returns, so the time series of PetroChina's returns is plotted as shown in Figure 7. It can be seen that China's oil returns are less volatile in some time periods and more volatile in others, a phenomenon that suggests that stock returns may have serial autocorrelation, i.e., there is a volatility aggregation phenomenon.

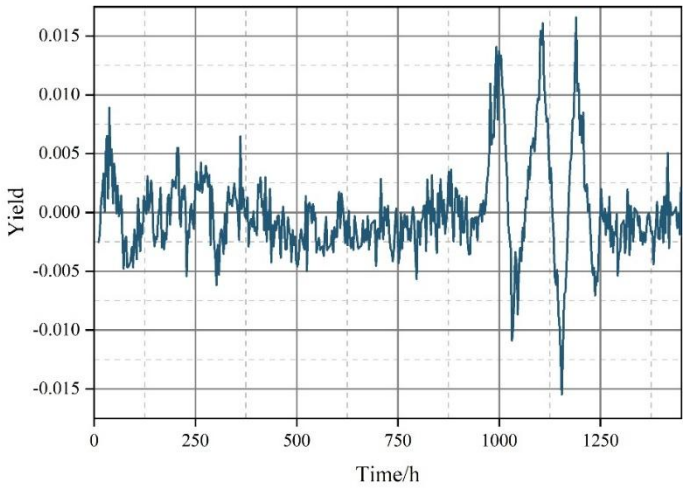


Figure 7: Time series of Chinese oil yields

To verify this conclusion, Ljung-Box-Pierce Q test and ARCH test are conducted. The ACF and PACF of China's oil yields and squared China's oil yields are shown in Tables 4 and 5, respectively. The serial correlation of the national oil yield series itself is not significant, but the correlation of its squared series is very significant, which indicates that the yields are not serially correlated but not mutually independent.

Table 4: ACF and PACF of China's oil yield

	AC	PAC		AC	PAC
1	0.013	0.013	13	0.033	0.049
2	-0.051	-0.051	14	0.006	0.029
3	0.043	0.046	15	-0.001	0.011
4	0.016	0.004	16	0.035	0.028
5	-0.005	-0.003	17	0.032	0.04
6	-0.035	-0.032	18	-0.05	-0.039
7	0.002	0.008	19	0.029	0.012
8	-0.016	-0.012	20	-0.023	-0.037
9	0.015	0.009	21	0.055	0.055
10	-0.056	-0.044	22	-0.04	-0.038
11	-0.018	-0.038	23	-0.011	-0.013
12	0.026	0.035	24	-0.031	0.026

Table 5: ACF and PACF of China's oil yield squared

	AC	PAC		AC	PAC
1	0.12	0.136	13	0.329	0.341
2	0.161	0.171	14	0.148	0.129
3	0.275	0.279	15	0.292	0.297
4	0.287	0.286	16	0.318	0.326
5	0.149	0.149	17	0.22	0.202
6	0.265	0.274	18	0.205	-0.206
7	0.267	0.271	19	0.179	0.178
8	0.301	0.304	20	0.268	0.267
9	0.127	0.131	21	0.142	0.148
10	0.237	0.246	22	0.242	-0.236
11	0.333	-0.317	23	0.27	-0.258
12	0.176	0.183	24	0.126	0.119

The Ljung-Box-Pierce Q-test is performed on the Chinese oil yield series and its squared series using Eviews software, and the test results of the lag terms of the 10th, 15th and 20th orders of the Chinese oil yield and the squared Chinese oil yield are shown in Tables 6 and 7, respectively. The test results show that at the 95% significance level, there is no significant autocorrelation in the first 20 orders of the yield series itself, however, the autocorrelation of its squared series is very significant.

Table 6: The Ljung-Box-Pierce Q test results of China's oil yield

Exponent	Pvalue	Stat	CriticalValue
Q(10)	0.9472	4.1177	18.3198
Q(15)	1.0002	6.2567	25.0019
Q(20)	0.835	13.915	31.4034

Table 7: The Ljung-Box-Pierce Q test results of China's oil yield squared

Exponent	Pvalue	Stat	CriticalValue
Q(10)	0	747.4699	18.3011
Q(15)	0	961.0048	24.9756
Q(20)	0	1203.1892	31.4144

The ARCH test on the series of China's oil yields using Eviews software is shown in Table 8. The ARCH effect of China's oil yield is significant.

Table 8: The ARCH test of China's oil yield square

Exponent	Pvalue	Stat	CriticalValue
Q(10)	0	44.1549	18.322
Q(15)	0	47.5765	24.998
Q(20)	0	62.5792	31.4012

In summary, the series of China's oil yields is indeed non-normal and characterized by sharp peaks and thick tails, so the improvements to the MC simulation method proposed in this chapter are reasonable.

3.3 Empirical demonstration of MC simulation method combining EEMD-LSTM models

The volatility predicted by the EEMD-LSTM combination model is brought into the Mento Carlo simulation method to calculate the January 4, 2024 PetroChina VaR value.

Programming the improved MC simulation method using Python software, the VaR value of the stock for the next January 3, 2024 was calculated as 2.33

For the need of model testing, we calculated the VaR for 255 trading days, i.e., on the basis of the above, we repeated the calculation 255 times using Python software to find out the daily VaR for the next 255 trading days at the confidence level of 95%, 97.5%, and 99%, respectively, and compared it with the actual loss to conduct the frequency of failure test, and the results are as shown in Table 9. From the test results, it can be seen that the Mento Carlo simulation method based on the EEMD-LSTM combination model only fails for 7 days at the higher confidence level of 99%, which falls within the rejection zone. And at all other confidence levels, the number of days to failure falls in the non-rejection zone. This suggests that, except at exceptionally high confidence levels, the estimation of VaR is still better when the combination model is added to the MC simulation method as well.

Table 9: The VaR model test results after an improved oil in China

Significance level	95%	97.5%	99%
Non-rejection interval	6<N<21	2<N<12	N<7
Actual days	20	10	7

4 Conclusion

In this paper, we firstly design an EEMD-LSTM based method for predicting financial market volatility. By using EEMD to decompose the time series, the intrinsic patterns of different periodical data can be better found. Moreover, compared with the traditional baseline forecasting model, the EEMD-LSTM model has a better fit, smaller root mean square error and average absolute error, and the model has a better forecasting effect on the volatility of financial assets in the financial market.

After that, the Copula function is used to improve the traditional Monte Carlo simulation (MC) method to calculate the VaR and realize the risk assessment of the financial market. The improved MC simulation method calculates the VaR values of financial assets with significant improvement in terms of calculation accuracy.

Finally combining the EEMD-LSTM model and the improved MC simulation method to calculate VaR. the number of days to failure is 7 only at the higher confidence level of 99%, and the number of days to failure falls in the non-rejection interval at all other confidence levels. It shows that the combination of the two is more effective in estimating VaR.

About the Authors

Qi Deng (born in November 1991), female, is a native of Fengqiu, Henan Province, China. She is a lecturer with a Master's degree, working in the Business School of Zhengzhou Technology and Business University. Her main research interests include digital economy and econometrics.

References

[1] Valenti, D., Fazio, G., & Spagnolo, B. (2018). Stabilizing effect of volatility in financial markets. Physical Review E, 97(6), 062307.

[2] Albulescu, C. T. (2021). COVID-19 and the United States financial markets' volatility. Finance research letters, 38, 101699.

[3] Bhowmik, R., & Wang, S. (2020). Stock market volatility and return analysis: A systematic literature review. Entropy, 22(5), 522.

[4] Atkins, A., Niranjan, M., & Gerding, E. (2018). Financial news predicts stock market volatility better than close price. The Journal of Finance and Data Science, 4(2), 120-137.

[5] Aït-Sahalia, Y., Li, C., & Li, C. X. (2021). Implied stochastic volatility models. The Review of Financial Studies, 34(1), 394-450.

[6] Tang, H. (2021, April). Stock prices prediction based on ARMA model. In 2021 International Conference on Computer, Blockchain and Financial Development (CBFD) (pp. i-iv). IEEE.

[7] Sun, H., & Yu, B. (2020). Forecasting financial returns volatility: a GARCH-SVR model. Computational Economics, 55(2), 451-471.

[8] Viljoen, H., Conradie, W. J., & Britz, M. M. (2022). The influence of different financial market regimes on the dynamic estimation of GARCH volatility model parameters and volatility forecasting. Studies in Economics and Econometrics, 46(3), 169-184.

[9] Kambouroudis, D. S., McMillan, D. G., & Tsakou, K. (2016). Forecasting stock return volatility: A comparison of GARCH, implied volatility, and realized volatility models. Journal of Futures Markets, 36(12), 1127-1163.

[10] Rubio, L., Palacio Pinedo, A., Mejía Castaño, A., & Ramos, F. (2023). Forecasting volatility by using wavelet transform, ARIMA and GARCH models. Eurasian Economic Review, 13(3), 803-830.

[11] Blazsek, S., & Mendoza, V. (2016). QARMA-Beta-t-EGARCH versus ARMA-GARCH: an application to S&P 500. Applied Economics, 48(12), 1119-1129.

[12] Ersin, Ö. Ö., & Bildirici, M. (2023). Financial volatility modeling with the GARCH-MIDAS-LSTM approach: The effects of economic expectations, geopolitical risks and industrial production during COVID-19. Mathematics, 11(8), 1785.

[13] Nilchi, M., & Farid, D. (2023). Modeling Price Dynamics and Risk Forecasting in Tehran Stock Exchange Market: Nonlinear and Non-gaussian Models of Stochastic Volatility. Financial Research Journal, 25(2), 275-299.

[14] 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.

[15] Chung, S. S., & Zhang, S. (2017). Volatility estimation using support vector machine: Applications to major foreign exchange rates. Electronic Journal of Applied Statistical Analysis, 10(2), 499-511.

[16] Kyoung-Sook, M. O. O. N., & Hongjoong, K. I. M. (2019). Performance of deep learning in prediction of stock market volatility. Economic Computation & Economic Cybernetics Studies & Research, 53(2).

[17] Koo, E., & Kim, G. (2022). A hybrid prediction model integrating garch models with a distribution manipulation strategy based on lstm networks for stock market volatility. IEEE Access, 10, 34743-34754.

[18] Son, B., Lee, Y., Park, S., & Lee, J. (2023). Forecasting global stock market volatility: The impact of volatility spillover index in spatial-temporal graph-based model. Journal of Forecasting, 42(7), 1539-1559.

[19] Wang, Y., & Zou, J. (2014). Volatility analysis in high-frequency financial data. Wiley Interdisciplinary Reviews: Computational Statistics, 6(6), 393-404.

[20] Kayim, F., & Yilmaz, A. (2022). Time series forecasting with volatility activation function. IEEE Access, 10, 104000-104010.

[21] Idrees, S. M., Alam, M. A., & Agarwal, P. (2019). A prediction approach for stock market volatility based on time series data. IEEE Access, 7, 17287-17298.

[22] Jarrah Mutasem. (2024). Long Short-Term Memory and Discrete Wavelet Transform based Univariate Stock Market Prediction Model. Journal of Information and Organizational Sciences,48(2),263-277.

[23] Department of Economics and Management North China Electric Power University Baoding China,Beijing Key Laboratory of New Energy and Low-Carbon Development North China Electric Power University Beijing China & Department of Economics and Management North China Electric Power University Baoding China. (2020). Forecasting the carbon price sequence in the Hubei emissions exchange using a hybrid model based on ensemble empirical mode decomposition. Energy Science & Engineering,8(8),2708-2721.

[24] Zongxuan Chai & Tingting Zheng. (2023). Systematic Risk Stress Prediction in Bond Market Based on EEMD-LSTM. Financial Engineering and Risk Management,6(11),

[25] Halil Ibrahim Gunduz ,Furkan Emirmahmutoglu& M. Eray Yucel. (2024). A New Look at Cross-Country Aggregation in the Global VAR Approach: Theory and Monte Carlo Simulation. Computational Economics,65(1),1-47.

[26] Chaoqiong Pan,Can Wang,Ziyan Zhao,Jinhao Wang & Zhaohong Bie. (2019). A Copula Function Based Monte Carlo Simulation Method of Multivariate Wind Speed and PV Power Spatio-Temporal Series. Energy Procedia,159,213-218.