

Comparative Evaluation of Machine Learning Methods for Financial Market Volatility Classification

Minghui Shi

Abstract—Financial market forecasting remains difficult because asset prices are strongly affected by randomness, noise, and limited short-term predictive information. Methods that aim to estimate precise future returns therefore often struggle to achieve consistently reliable results. To address this issue, this study reformulates the prediction task as a classification problem that distinguishes between different levels of market volatility, taking advantage of the persistence and clustering commonly observed in volatility dynamics. Ten machine learning methods are evaluated under a unified experimental framework, covering linear classifiers, support vector machines, instance-based approaches, decision trees, and ensemble techniques. Under the current experimental setting, all tested models achieve classification accuracies above 80%, and Gradient Boosting records the highest accuracy of 85.05% together with the highest F1-score. The linear classifiers nevertheless remain competitive: Logistic Regression ranks second in accuracy, only 0.003 below the leading ensemble method, and Linear SVM also places within the top four. The comparison also reveals a clear precision–recall trade-off, since RBF SVM attains the highest precision while Ridge Classifier attains the highest recall. These results indicate that, for the dataset and experimental configuration considered here, volatility classification can be addressed with comparable effectiveness by models of substantially different complexity, so that model selection should take several evaluation measures into account rather than accuracy alone.

Key Words—Machine Learning, Financial Markets, Volatility Prediction, Classification, Ensemble Learning, Investment Decision Support

I. INTRODUCTION

Financial markets exhibit highly dynamic behavior shaped by nonlinear relationships, feedback mechanisms, and substantial stochastic variation. Changes in asset prices may reflect numerous sources of information, ranging from macroeconomic conditions and investor expectations to geopolitical developments and market microstructure effects. As a result, forecasting short-horizon market movements remains a difficult problem in quantitative finance. Although machine learning and computational forecasting techniques have advanced considerably, the weak signal embedded in noisy

financial observations continues to limit the accuracy of return prediction.

Conventional financial forecasting techniques, including linear regression and classical time-series models, are commonly developed under assumptions such as stationarity, normally distributed observations, or relatively stable statistical relationships. In practice, financial series often depart from these assumptions because market behavior can change across structural breaks, market regimes, and periods of varying volatility, motivating regime-switching formulations such as Markov models of market ambiguity [1]. Machine learning models provide greater flexibility for representing complex and nonlinear relationships, but this flexibility does not eliminate the fundamental difficulty of predicting short-term returns. When price changes contain only limited exploitable information, increasing model complexity alone may not lead to substantial improvements in forecasting performance.

An alternative is to shift the prediction objective from exact price or return estimation toward the identification of future volatility conditions. Instead of determining the precise direction or magnitude of subsequent price movements, volatility classification separates market observations into relatively high- and low-volatility states. This formulation is motivated by the well-established persistence of volatility, whereby elevated volatility is often followed by further periods of elevated volatility and calmer market conditions may likewise persist over time. Such temporal structure, which reflects the stylized fact that volatility tends to cluster rather than evolve as an independent process, provides a potentially more consistent target for machine learning-based volatility classification [2, 3].

This classification setting also enables the use of information extracted from historical market observations through feature engineering. Rather than relying solely on raw returns, the input representation can incorporate rolling volatility statistics, moving averages, momentum-related variables, and lagged returns. These derived features describe different aspects of recent market behavior, including price trends, short-term fluctuations, and temporal dependence, thereby providing a richer representation of market conditions for predictive modeling.

Based on this formulation, the present study evaluates ten machine learning classifiers drawn from several major modeling families, including linear methods, support vector machines, instance-based learning, decision trees, and ensemble approaches. Evaluating models with substantially different levels of structural complexity makes it possible to compare their ability to distinguish between low- and high-volatility

Copyright: ©2026 by the authors. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Manuscript received August 23, 2026; revised September 15, 2026; accepted October 7, 2026. This article was recommended for publication by Associate Editor Shujin Qin upon evaluation of the reviewers' comments.

M. Shi is with the Artificial Intelligence and Software College, Liaoning Petrochemical University, Fushun 113001, China (e-mail: fnex1024s@163.com).

Corresponding author: Minghui Shi.

conditions and to examine whether more sophisticated nonlinear techniques provide meaningful improvements over simpler alternatives.

Beyond the direct comparison of classification performance, the study also considers the role of feature construction and data representation in the interpretation of the results. Because all ten classifiers are evaluated using the same feature representation, the comparison allows differences in classification performance to be examined without introducing variation in the input variables across models. It should be emphasized that the study is designed as a comparative evaluation performed under a single experimental configuration; it does not propose a new learning algorithm, and it is not intended to establish that volatility classification is generally superior to alternative forecasting formulations or that the models examined here would remain effective in other markets or periods. The overall objective is therefore to assess how the ten selected classifiers perform on the volatility classification task within the present experimental setting and to identify the trade-offs that distinguish them.

II. RELATED WORK

Machine learning techniques have been widely investigated in financial applications, including market forecasting, risk assessment, and algorithmic trading [4]. Earlier studies were largely based on conventional statistical and econometric approaches, such as linear regression, autoregressive integrated moving average (ARIMA), and related time-series models. These methods offer relatively transparent model structures and are often easy to interpret. However, their ability to represent nonlinear relationships and complex temporal patterns in financial data is generally limited.

As machine learning methods became more widely adopted, greater attention was given to nonlinear approaches. Support Vector Machines were among the first such methods to be adopted for financial classification and regression tasks, followed by tree-based ensemble models such as Random Forest and Gradient Boosting. These techniques have since been applied to market forecasting [5, 6], risk assessment, and the prediction of volatility-related quantities [7]. They are capable of representing more complex interactions among input variables and, in the case of ensemble methods, improving predictive stability by combining multiple learners. Nevertheless, their performance remains strongly influenced by the quality of the input representation [8] and by the effectiveness of the feature engineering pipeline [9]. This issue is especially important in financial prediction, where useful information is often obscured by substantial market noise.

Directly forecasting future returns remains one of the major difficulties reported in financial prediction research. From the perspective of the Efficient Market Hypothesis, newly available information is rapidly incorporated into asset prices, which limits the possibility of consistently predicting subsequent price changes [10]. Consequently, researchers have increasingly explored alternative prediction targets, including the forecasting of realized or implied volatility [11], the classification of price direction, the identification of market

regimes and volatility states, and the detection of anomalous market behaviour [12]. Compared with exact return estimation, these formulations can provide more structured prediction targets and may correspond more closely to practical financial decision-making tasks.

Recent research has also introduced deep learning methods for financial time-series modeling. Long Short-Term Memory (LSTM) networks [13] have been used to forecast realized volatility [14] and to exploit commonality across intraday observations [15], while recurrent architectures and other sequence models have been applied to volatility forecasting under rough volatility dynamics [16] and to hybrid forecasting designs [17]. Such architectures are designed to learn sequential relationships and longer-range temporal dependencies directly from time-series observations. Although these models provide greater representational capacity, reported results are not uniformly superior to those of conventional machine learning methods. Their performance can vary considerably depending on preprocessing procedures, feature representation, data quality, and training stability. Under noisy or relatively limited datasets, simpler learning algorithms may therefore remain competitive.

Hybrid and ensemble modeling has emerged as another important direction in financial forecasting research. Such approaches combine statistical techniques, machine learning algorithms, stacked learners [18], or heterogeneous financial indicators [19] in order to exploit complementary modeling capabilities and reduce the weaknesses of individual methods. At the same time, researchers have increasingly considered information beyond historical price variables. Textual and news-based data have been incorporated through financial word embeddings [20] and sentiment-related variables [21], while macroeconomic and cross-market indicators have been used to capture changes in market conditions [22].

Beyond financial applications, learning-based and optimization-based methods have also been applied to complex industrial decision problems, including knowledge-guided hyper-heuristics for distributed manufacturing scheduling [23], data-driven modeling of human-robot collaborative production systems [24], and reinforcement learning for assembly line balancing [25]. Such studies illustrate the breadth of the methodological toolkit considered in this work, although their objectives, data characteristics, and evaluation criteria differ substantially from those of volatility classification.

The studies reviewed above differ in their modeling assumptions, input representations, prediction targets, and evaluation criteria, which makes their reported results difficult to compare directly. Motivated by this gap, the present study evaluates a broad collection of machine learning classifiers within a common experimental setting. Rather than restricting the analysis to a single family of algorithms, models with different learning mechanisms and levels of complexity are compared using consistent preprocessing procedures, evaluation criteria, and experimental conditions. This design supports a direct comparison of their relative effectiveness for market volatility classification under identical conditions.

III. DATASET

The experiments in this study are based on the *Market Trend Analysis Dataset* obtained from Kaggle [26]. The dataset contains historical financial market observations organized in time-series form and provides numerical variables describing price movements, trading activity, and technical market indicators. Its chronological structure makes it suitable for preserving temporal order during preprocessing, model training, and subsequent evaluation.

The original data include fundamental market variables such as open price, close price, high price, low price, and trading volume, together with the trading activity measures commonly used in empirical studies of volatility. In addition to these raw observations, several technical indicators are provided, including daily return percentage, volatility range, moving averages, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Bollinger Band values. Together, these variables characterize multiple aspects of market behavior, including price level, trend, momentum, and volatility.

For the feature engineering process, the close price was used as the principal reference series. Additional variables were derived from this price series in order to represent short-term returns, rolling statistical behavior, and temporal dependence in market movements. These transformations, which follow the standard practice of constructing indicator-based inputs for data-driven volatility models [27], provided a richer description of recent market conditions and produced input variables that could be directly used by the machine learning classifiers.

Before model development, the dataset underwent a series of preprocessing operations. Non-numeric variables were removed, missing observations were handled, and infinite values arising from feature calculations were eliminated. The remaining observations were then arranged in chronological order so that the temporal structure of the financial series was maintained throughout the experiment. Following data cleaning and feature construction, the processed dataset contained 29,975 observations and 32 columns.

For the final classification task, 30 variables were retained as model inputs. The future volatility measure and the binary volatility class generated from that measure were excluded from the feature set and used only for target construction. The resulting inputs therefore consisted of both original market variables and engineered technical features, providing the information used to train and evaluate the machine learning models considered in this study.

IV. METHODOLOGY

A. Feature Engineering

Feature construction plays an important role in representing the temporal and statistical characteristics of market behavior. In this study, multiple groups of derived variables are generated from historical price information, including:

- Returns calculated over 1-day, 3-day, and 5-day horizons
- Moving averages computed using 5-day, 10-day, and 20-day windows

- Deviations of current prices from their corresponding moving averages
- Momentum-based indicators
- Lagged return variables
- Rolling measures of market volatility

These engineered variables are intended to describe different patterns in market movements, particularly short-term trend persistence and potential mean-reversion behavior.

B. Target Definition

The prediction target is defined according to future market volatility. Because volatility is known to be persistent and to cluster in time rather than to evolve as an independent process [28, 29], a rolling standard deviation of returns is first calculated over a moving window and then shifted forward in time, so that the volatility measure associated with a given observation refers to a subsequent interval rather than to the observation itself. The volatility measure is computed as

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n (r_i - \bar{r})^2} \quad (1)$$

where r_i denotes an observed return within the rolling window, $\bar{r}$ represents the corresponding mean return, and n is the number of observations included in the calculation.

The continuous future volatility measure is subsequently converted into a binary classification target using its median as the decision threshold. In the present experiment, the resulting volatility threshold is 0.024289086264440325. Because the threshold is defined in terms of the volatility scale of this dataset, the resulting class boundary is specific to the data and the rolling window used here. The two classes are defined as follows:

- 0: Low volatility
- 1: High volatility

This transformation converts the forecasting problem from continuous volatility estimation into the identification of low- and high-volatility market conditions.

C. Data Splitting Strategy

To preserve the temporal structure of the financial data, the processed observations are divided chronologically rather than through random sampling. Earlier observations are assigned to the training set, whereas later observations are reserved for testing. This arrangement is intended to reproduce a forecasting setting in which models are trained using historical information and subsequently evaluated on later, unseen observations, and it avoids the temporal mixing that random sampling would introduce. It should be noted that a chronological split, by itself, does not remove every possible form of temporal dependence between the input variables and the target, and the results reported in this study are therefore interpreted as the performance of the classifiers within this particular design. The final training set contains 23,980 samples, and the testing set contains 5,995 samples.

V. ALGORITHMS

A total of ten machine learning classifiers from several major model families are included in the comparative experiment.

A. Linear Models

Logistic Regression and Ridge Classifier are selected as representative linear approaches and serve as baseline methods in the comparison. Both models construct decision functions based primarily on linear relationships between the input variables and the volatility class.

B. Support Vector Machines

Two Support Vector Machine configurations are considered: Linear SVM and RBF SVM. Linear SVM separates the two volatility classes using a linear decision boundary, whereas RBF SVM employs a radial basis function kernel to represent nonlinear separation patterns in the feature space [30].

C. Instance-Based Learning

K-Nearest Neighbors (KNN) is included as an instance-based classification method. Rather than explicitly learning a global decision function, KNN determines the class of a sample according to the labels of nearby observations with similar feature characteristics.

D. Tree-Based Models

Decision Tree is evaluated as a representative tree-based classifier. The model recursively partitions the input feature space according to a sequence of decision rules, producing an interpretable structure for assigning observations to volatility classes.

E. Ensemble Methods

Four ensemble algorithms are examined: Random Forest, Extra Trees, Gradient Boosting, and HistGradientBoosting. Random Forest aggregates decorrelated trees grown on bootstrap samples [31], whereas Gradient Boosting and its histogram-based variant build trees sequentially so that each new learner corrects the residual error of the previous ensemble [32]. These methods construct predictions using collections of decision trees rather than relying on a single tree. By combining information from multiple learners, they are designed to improve predictive stability and capture more complex relationships among the input features.

VI. EVALUATION METRICS

The predictive performance of the models is assessed using four commonly adopted classification measures: accuracy, precision, recall, and F1-score. Considering these metrics together provides a more complete characterization of classification behavior than relying on accuracy alone. This is particularly important in financial applications, where different types of classification errors may have different practical consequences.

Accuracy represents the proportion of observations that are assigned to the correct volatility class. Although it provides

a straightforward measure of overall classification performance, it does not distinguish between false-positive and false-negative predictions. Consequently, accuracy alone may provide an incomplete assessment when the class distribution is uneven or when different prediction errors carry different levels of importance.

Precision evaluates the reliability of high-volatility predictions by measuring the proportion of samples predicted as high volatility that actually belong to the high-volatility class. A higher precision value therefore corresponds to fewer false-positive signals. This consideration is relevant in financial applications because unnecessary high-volatility warnings may result in overly conservative adjustments to trading decisions or risk exposure.

Recall, also referred to as sensitivity or the true positive rate, measures the proportion of actual high-volatility observations that are successfully detected by a classifier. It therefore reflects the ability of a model to recognize genuinely volatile market conditions. A model with high recall is less likely to overlook high-volatility periods, which can be desirable in risk-sensitive applications. However, an increase in recall may also be accompanied by a larger number of false-positive predictions and consequently lower precision.

The F1-score combines precision and recall through their harmonic mean. It provides a single measure of the balance between identifying high-volatility observations and limiting false-positive predictions. This metric is particularly informative when neither precision nor recall should dominate the evaluation.

Taken together, the four measures make it possible to compare not only the overall classification ability of each algorithm but also its behavior in detecting high-volatility conditions. It should be emphasized that all four measures quantify the statistical behavior of the classifiers with respect to the constructed volatility labels. They do not, by themselves, measure the economic value of a prediction, because they do not account for transaction costs, position sizing, or the payoff structure of any particular trading or risk-management rule. A high value of accuracy, precision, recall, or F1-score therefore indicates a closer correspondence between predicted and observed volatility states, and not a corresponding improvement in financial outcome.

VII. RESULTS AND DISCUSSION

The experimental analysis compares ten machine learning algorithms for binary market volatility classification, with class 0 corresponding to low-volatility conditions and class 1 representing high-volatility conditions. Following feature engineering and data cleaning, the processed dataset contains 29,975 observations and 32 columns. The resulting class distribution is nearly balanced, consisting of 49.97% low-volatility samples and 50.03% high-volatility samples. A chronological train-test split produces 23,980 training observations and 5,995 testing observations. The evaluated models include Logistic Regression, Ridge Classifier, Linear SVM, RBF SVM, KNN, Decision Tree, Random Forest, Extra Trees, Gradient Boosting, and HistGradientBoosting.

The approximately balanced class distribution of the processed dataset reduces the possibility that overall classification performance is dominated by one volatility class. In classification problems with strongly unequal class frequencies, high accuracy can sometimes result from repeatedly predicting the majority class. In the present experiment, the similar proportions of low- and high-volatility observations make accuracy, precision, recall, and F1-score more informative for comparing model behavior across the two classes.

The chronological training and testing procedure is also important for maintaining the temporal ordering of the financial observations. Rather than randomly mixing earlier and later samples, the models are trained using historical observations and subsequently evaluated using later, previously unseen observations. This design more closely resembles a forecasting setting, in which future market information would not be available when the models are trained, and it avoids the temporal mixing associated with random data splitting. As noted in Section IX, this arrangement does not by itself eliminate every possible source of temporal dependence between the inputs and the target.

The ten algorithms were selected to represent distinct learning paradigms, ranging from linear decision boundaries, through margin-based and instance-based classification, to single-tree and ensemble-based partitioning. Their structures and levels of complexity differ substantially, which allows their behavior to be compared under identical experimental conditions.

The results indicate that all evaluated methods achieve relatively high classification performance, with accuracy exceeding 80% for every model. The differences among the algorithms are nevertheless modest, which suggests that the predictive information available in the current feature representation can be utilized by several different model families. The competitive performance of Logistic Regression, Ridge Classifier, and Linear SVM relative to more complex ensemble models suggests in addition that a considerable part of the structure present in the current feature representation can be captured using relatively simple decision functions.

Ensemble methods are also competitive in this comparison. In particular, Gradient Boosting records the highest accuracy and F1-score, while Extra Trees, HistGradientBoosting, and Random Forest achieve results within a similar range. These margins are small, however, and no statistical test of significance is applied to them in this study. Some nonlinear models, including KNN and RBF SVM, also rank below the linear classifiers in terms of accuracy. The results therefore do not establish a consistent advantage of nonlinear models over linear alternatives under the present setting, and they indicate that greater model complexity does not automatically translate into substantially better classification performance in this experiment.

The comparison also reveals meaningful differences between precision and recall. Ridge Classifier produces the highest recall and therefore identifies a larger proportion of actual high-volatility observations, although this behavior is accompanied by lower precision. RBF SVM, in contrast, achieves the highest precision and therefore generates fewer

incorrect high-volatility predictions relative to its total number of positive predictions. Such differences are relevant in financial applications because false positives and false negatives may lead to different practical consequences. Missing a genuinely volatile period may increase risk exposure, whereas incorrectly signaling high volatility may result in unnecessarily cautious decisions.

Overall, the experimental results show that, within the present experimental setting, the ten machine learning algorithms are capable of distinguishing between low- and high-volatility observations with broadly similar levels of performance. The relatively narrow differences across models further indicate that model selection should not depend exclusively on overall accuracy. Precision, recall, and their associated trade-offs should also be considered according to the requirements of the intended financial application.

TABLE I Model Comparison Across Ten Algorithms

Algorithm	Accuracy	Precision	Recall	F1
Gradient Boosting	0.8505	0.8431	0.8824	0.8623
Logistic Regression	0.8475	0.8350	0.8881	0.8607
Extra Trees	0.8469	0.8394	0.8796	0.8590
Linear SVM	0.8465	0.8294	0.8947	0.8608
HistGradientBoosting	0.8447	0.8386	0.8758	0.8568
RBF SVM	0.8445	0.8552	0.8509	0.8531
Ridge Classifier	0.8430	0.8107	0.9186	0.8613
Random Forest	0.8425	0.8370	0.8733	0.8547
KNN	0.8347	0.8419	0.8475	0.8447
Decision Tree	0.8257	0.8270	0.8491	0.8379

As shown in the model comparison, Gradient Boosting achieves the highest accuracy and F1-score among the ten algorithms. RBF SVM records the highest precision, whereas Ridge Classifier obtains the highest recall. Despite these differences, every evaluated classifier achieves an accuracy greater than 80%.

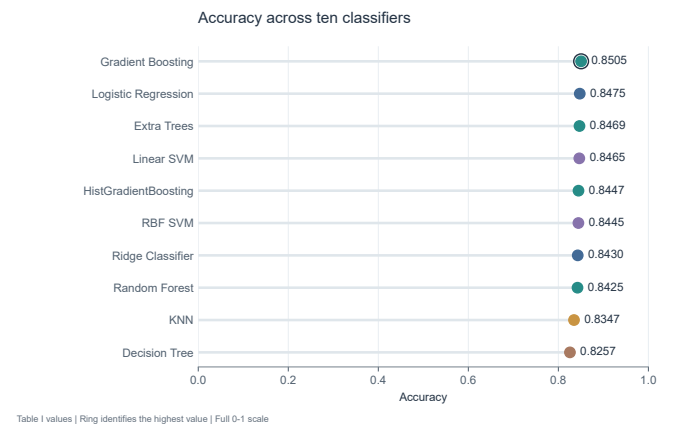


Fig. 1. Accuracy comparison across ten algorithms.

The accuracy results are concentrated within a relatively narrow interval, with most classifiers achieving values between approximately 0.83 and 0.85. Gradient Boosting ranks first with an accuracy of 0.8505, but its advantage over Logistic Regression, Extra Trees, and Linear SVM is small. The limited separation between these methods indicates that the

classification performance obtained in this experiment is not determined solely by algorithmic complexity, as both relatively simple and more sophisticated models achieve similar levels of accuracy.

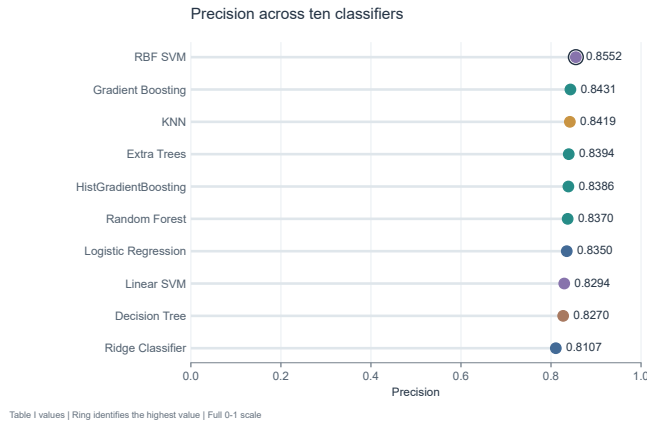


Fig. 2. Precision comparison across ten algorithms.

The precision comparison focuses on the reliability of predictions assigned to the high-volatility class. RBF SVM obtains the highest precision, indicating that a larger proportion of its high-volatility predictions correspond to actual high-volatility observations. Ridge Classifier records a lower precision value, reflecting a greater proportion of false-positive predictions. This distinction is relevant when incorrect volatility warnings may lead to unnecessary changes in investment or risk-management decisions.

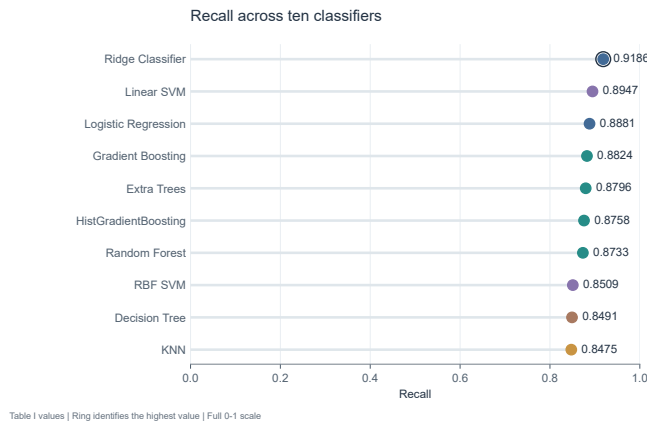


Fig. 3. Recall comparison across ten algorithms.

Recall measures the ability of each classifier to detect observations that actually belong to the high-volatility class. Ridge Classifier achieves the highest recall and therefore identifies the largest fraction of actual high-volatility samples. Its high recall, however, is accompanied by comparatively lower precision. Gradient Boosting and Linear SVM also achieve high recall while maintaining a more balanced relationship with precision.

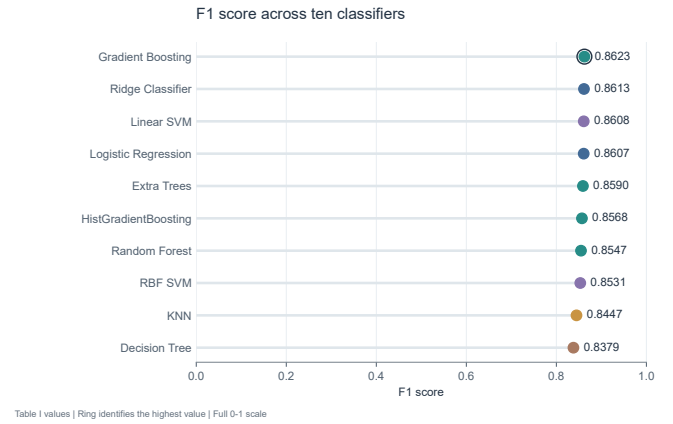


Fig. 4. F1 score comparison across ten algorithms.

The F1-score summarizes the balance between precision and recall. Gradient Boosting records the highest F1-score, indicating the strongest overall balance between detecting high-volatility observations and limiting false-positive classifications. Logistic Regression, Linear SVM, and Ridge Classifier also obtain competitive F1 values, showing that several model families provide a favorable balance between the two measures.



Fig. 5. Grouped metric comparison of accuracy, precision, recall, and F1 score.

The grouped comparison provides an integrated view of accuracy, precision, recall, and F1-score for all ten algorithms. Most models display relatively consistent values across the four evaluation measures, with only moderate variation among algorithms. Gradient Boosting demonstrates a strong balance across the metrics, whereas Ridge Classifier is characterized by particularly high recall together with comparatively lower precision.

The same comparison further illustrates the precision-recall trade-off observed among the classifiers. Gradient Boosting maintains relatively strong values for both measures, while Ridge Classifier favors the identification of high-volatility observations more aggressively. As a result, Ridge Classifier captures more true positive cases but also produces a larger proportion of false-positive predictions.

Classification outcomes by actual volatility class



Fig. 6. Combined confusion matrix visualization for top-performing models (Gradient Boosting, Logistic Regression, Extra Trees, and Linear SVM).

The confusion matrices provide a more detailed view of the classification outcomes of four of the strongest-performing methods. Gradient Boosting correctly identifies 2,293 low-volatility samples and 2,806 high-volatility samples, while 522 low-volatility observations and 374 high-volatility observations are incorrectly classified. Logistic Regression correctly predicts 2,257 low-volatility samples and 2,824 high-volatility samples. Extra Trees correctly identifies 2,280 low-volatility observations and 2,797 high-volatility observations, while Linear SVM correctly classifies 2,230 low-volatility samples and 2,845 high-volatility samples. These results show that the leading models are able to correctly classify substantial numbers of observations from both volatility classes.

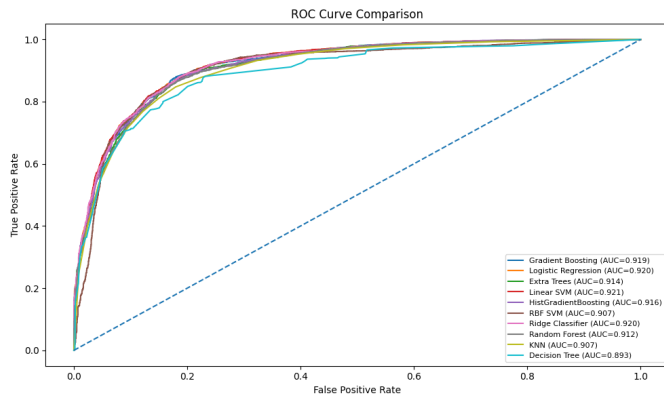


Fig. 7. ROC curve comparison across all models.

The ROC curves illustrate the relationship between the true positive rate and false positive rate across different classification thresholds. All evaluated models perform above the random-classification baseline, indicating that they possess meaningful ability to discriminate between the two volatility classes. Among the evaluated algorithms, Linear SVM produces the highest AUC, followed closely by Logistic Regression and Ridge Classifier. The strong performance of these linear methods suggests that linear decision functions are effective for separating the volatility classes under the feature representation used in this experiment.

Nearly balanced volatility classes



Percentages and total n from manuscript Section VII | Exact class counts were not supplied

Fig. 8. Class distribution for low-volatility and high-volatility samples.

The class distribution shows that low- and high-volatility samples occur in nearly equal proportions in the processed dataset. Because neither class strongly dominates the other, the reported accuracy values are less likely to be driven simply by majority-class predictions. The balanced distribution also supports the joint interpretation of accuracy, precision, recall, and F1-score when comparing the ten classifiers. It should be noted that this near balance refers to the complete processed dataset, whereas the class proportions in the chronological test set differ: of the 5,995 test observations, 2,815 belong to the low-volatility class and 3,180 to the high-volatility class, as reflected in the confusion matrices in Fig. 6. The larger class accounts for about 53% of the test set, so a majority-class predictor would already reach an accuracy of approximately 0.53, and precision and recall are likewise not computed over equally sized groups of positive and negative observations.

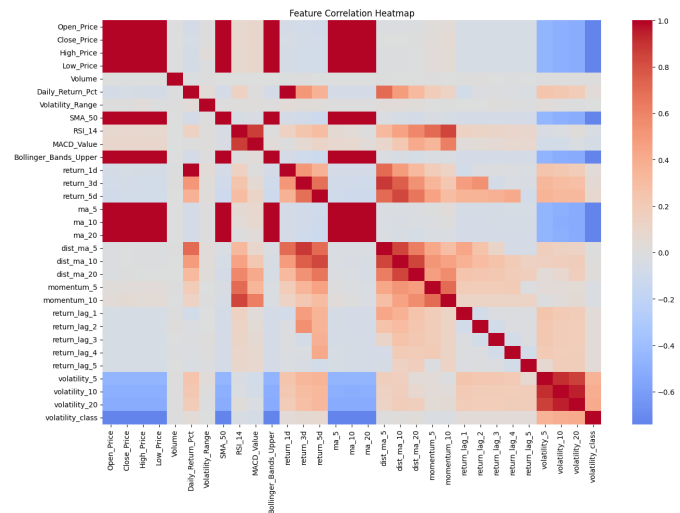
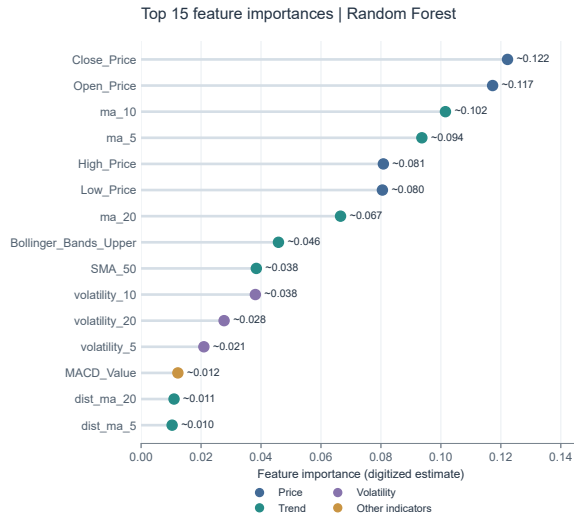


Fig. 9. Combined feature correlation heatmap visualization.

The feature correlation heatmap reveals strong relationships among several price-related variables, including open, close, high, and low prices, as well as moving averages and Bollinger Band indicators. Several volatility-oriented features show comparatively strong associations with the volatility

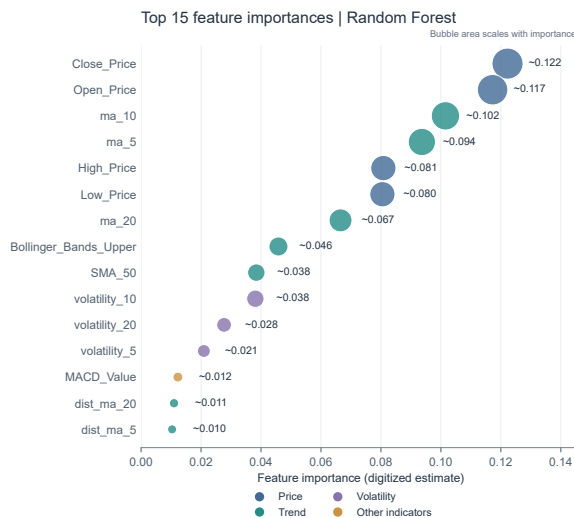
class. These patterns are consistent with the use of rolling volatility measures, lagged returns, and momentum-related variables as inputs to the classification models. Correlation coefficients describe linear association between variable pairs only; they do not indicate that a feature contributes causally to the classification outcome, and the strong mutual correlation among the price variables means that their individual roles cannot be separated from this summary alone.



~ denotes an estimate recovered from raster bar length; original model outputs are needed for exact values.

Fig. 10. Top 15 feature importances for Random Forest.

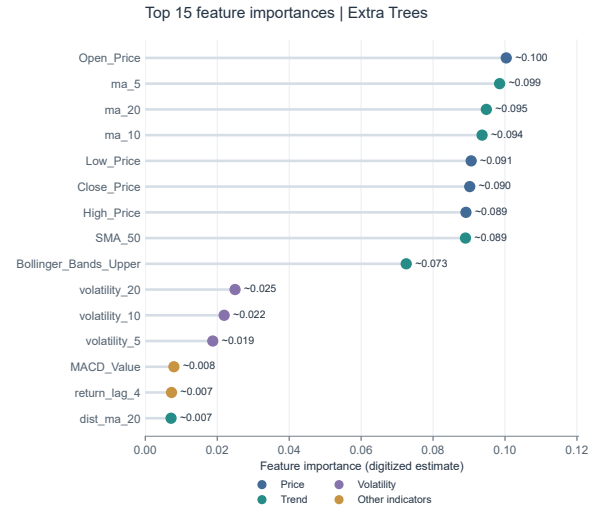
The Random Forest importance ranking identifies several price-based, moving-average, and volatility-related variables among the most influential features used by the classifier. This distribution suggests that the model makes use of information from both original market variables and engineered indicators rather than relying exclusively on a single type of feature.



~ denotes an estimate recovered from raster bar length; original model outputs are needed for exact values.

Fig. 11. Second Random Forest feature importance visualization.

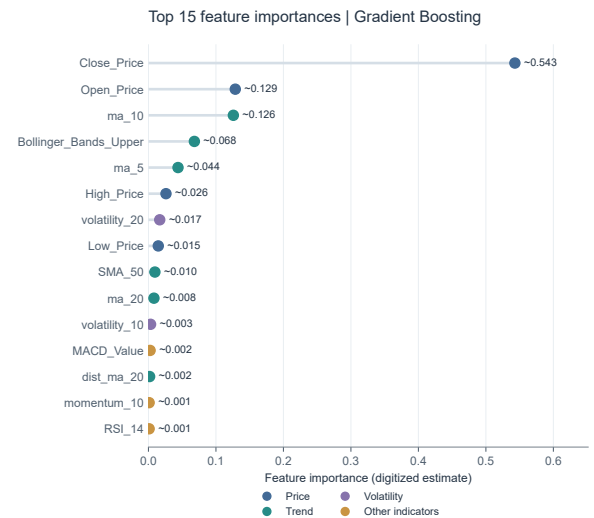
The second Random Forest feature-importance visualization presents a similar pattern, again emphasizing variables related to price behavior and technical indicators. The similarity between the two visualizations indicates that comparable groups of variables contribute substantially to the model's classification decisions.



~ denotes an estimate recovered from raster bar length; original model outputs are needed for exact values.

Fig. 12. Top 15 feature importances for Extra Trees.

The Extra Trees feature-importance results show that predictive importance is distributed across multiple price- and volatility-related variables. Rather than depending on a single dominant predictor, the model draws information from several components of the feature representation when separating low- and high-volatility observations.



~ denotes an estimate recovered from raster bar length; original model outputs are needed for exact values.

Fig. 13. Top 15 feature importances for Gradient Boosting.

The Gradient Boosting feature-importance ranking is more concentrated than the corresponding rankings for Random Forest and Extra Trees. The Close_Price feature receives

a noticeably higher importance score than the remaining predictors, while the importance assigned to other features is distributed across price-related variables and engineered technical indicators. Because several price variables in the dataset are strongly correlated with one another, the measured importance may be shared among them rather than attributed exclusively to one variable, and the leading position of a particular price feature should not be read as evidence that it carries a distinct predictive signal. It should also be noted that importance scores in tree-based models summarize how frequently and how effectively a variable is used for splitting during training; they are not measures of causal effect and do not by themselves account for the differences in classification performance reported in Table I.

VIII. EXTENDED ANALYSIS

Several broader observations can be drawn from the comparative results. First, the strong performance of the linear classifiers indicates that the relationship between the engineered financial indicators and the volatility classes can, to a considerable extent, be represented using linear decision boundaries. Logistic Regression, Linear SVM, and Ridge Classifier all achieve results close to those of the best-performing ensemble approaches.

Second, the ensemble methods are also competitive in this comparison. Gradient Boosting obtains the highest accuracy and F1-score, while Extra Trees and HistGradientBoosting achieve results within a similar range. As discussed above, however, the margins separating the ensemble methods from the linear classifiers are small, and some nonlinear models rank below the linear ones in accuracy. The present results therefore do not establish that nonlinear interactions among price variables, moving averages, momentum indicators, and volatility-related features provide information beyond that captured by linear decision boundaries. Distinguishing between these two possibilities would require a more targeted experimental design than the one adopted here.

Third, the overall performance differences among the evaluated algorithms remain relatively small. Gradient Boosting, the most accurate classifier, achieves an accuracy of 0.8505, compared with 0.8257 for Decision Tree. The narrow range across models suggests that the predictive information available in the current feature representation can be utilized by classifiers of different levels of complexity, and that increasing algorithmic complexity alone does not account for the observed ranking. It should be noted that this study does not include an ablation over the feature groups described in Section IV, and consequently the individual contribution of specific feature subsets, or of feature construction relative to the original market variables, is not separately quantified. The comparable performance of the different model families indicates that the information present in the current representation can be exploited by both simple and complex classifiers; it does not by itself measure how much of that performance is attributable to the engineered features.

The results also illustrate the behaviour of volatility classification as an alternative to the direct prediction of exact returns,

although the two formulations are not compared experimentally in this study. Rather than estimating the precise future direction or magnitude of price changes, the classification framework identifies whether subsequent market conditions correspond to a relatively low- or high-volatility state. This produces a discrete representation of future market uncertainty that may serve as an input to subsequent financial analysis, while the economic usefulness of that representation is not evaluated here.

A further consideration is the construction of the prediction target. The binary label is obtained by thresholding a rolling volatility measure that is shifted forward in time, and financial volatility is itself strongly persistent. As a result, the inputs and the target share a common underlying dynamic, and part of the classification performance reported above may originate from this temporal dependence rather than from information that would be available for an unrelated prediction problem. The reported values should therefore be read as the performance of the ten classifiers under this particular target definition, and not as an estimate of how well the same models would perform for volatility prediction in general.

IX. LIMITATIONS

Despite the favorable classification results, the study has several limitations. First, the dataset does not capture the full range of information that can influence real financial markets. Although historical price variables and technical indicators are included, important external drivers of volatility are absent, even though previous studies have shown that macroeconomic conditions, cross-market spillovers [33], and news-based sentiment signals can contribute additional predictive information. Changes in interest rates, inflation, employment conditions, and central bank policies can affect market behavior, as can firm-specific information such as earnings announcements and financial guidance. Geopolitical events, regulatory developments, market crises, and changes in liquidity may also generate substantial volatility. Because these factors are not represented in the current dataset, the models cannot capture all possible sources of variation in market conditions.

A second limitation concerns the construction of the prediction target. Future volatility is derived from historical price information using rolling volatility calculations. Since volatility commonly exhibits temporal persistence and clustering, volatility-related features may contain considerable information about subsequent volatility states. Consequently, part of the observed classification performance may arise from this temporal dependence rather than from entirely independent predictive patterns. This characteristic should be considered when interpreting the reported performance and its applicability to less structured forecasting problems. The interpretation of the classification results is also influenced by the construction of the volatility target and the temporal persistence of financial volatility: because the target is a specific function of the rolling volatility estimate and its forward shift, the reported performance is tied to this particular definition, and different window lengths or forward horizons could lead to different results.

A related consideration concerns the evaluation baseline. The present experiment compares ten machine learning classifiers with one another under a common experimental setting and does not include a separate comparison against simple persistence-based forecasting rules. Since financial volatility exhibits clustering, a rule that carries the prevailing volatility state forward could already account for part of the classification accuracy reported here. The results therefore establish the relative performance of the selected algorithms within this experimental setting, but they do not independently quantify the incremental predictive benefit of machine learning over simpler benchmark approaches.

The analysis is also based on a single dataset and a single chronological partition of that dataset. Volatility dynamics may differ considerably across financial instruments, market periods, and economic environments. Equity, commodity, and cryptocurrency markets, for example, can exhibit different volatility dynamics and responses to changing market conditions, as documented for cryptocurrencies [34], for oil futures subject to cross-market spillovers, and for digital assets during periods of market stress. Without evaluation on additional datasets or across different market regimes, the extent to which the reported results generalize beyond the current experimental setting remains uncertain. Because the models are trained and tested on one particular division of the data, the stability of the ranking in Table I across different time periods has likewise not been examined, and the ordering reported here should be understood as the outcome of this division rather than as an established ordering that would hold for earlier or later periods.

Another limitation is that model effectiveness is evaluated primarily through classification measures such as accuracy, precision, recall, and F1-score. The experiment does not directly examine whether these predictions improve performance within an actual trading or investment strategy. In practical applications, transaction costs, bid-ask spreads, slippage, execution delays, liquidity constraints, and available capital can substantially affect financial outcomes, and studies that translate volatility forecasts into timing strategies or that quantify the economic value of volatility predictability show that statistical accuracy alone is not sufficient. Classification performance alone therefore does not establish whether a model would produce economic benefits after being incorporated into a complete investment system.

The models are also evaluated without extensive hyperparameter optimization. The experimental configurations are based on reasonable default or manually selected settings rather than systematic tuning procedures. More comprehensive approaches, including grid search, random search, or Bayesian optimization, could potentially alter the performance of individual algorithms. The current results should therefore be interpreted as a comparison under the implemented configurations rather than as the maximum achievable performance of each model family.

Finally, model interpretability is not investigated in detail. Although linear classifiers and decision trees provide relatively transparent decision structures, ensemble and nonlinear models can be more difficult to interpret. In financial decision-making, understanding the factors that contribute to model predictions

can be important for transparency, trust, and accountability. The absence of a detailed explainability analysis limits the extent to which the internal prediction mechanisms of the more complex models can be examined.

These limitations identify several opportunities for extending the current study. Broader data sources, evaluation across multiple market environments, consideration of realistic financial constraints, systematic model tuning, and more detailed interpretability analysis would provide a more comprehensive assessment of machine learning-based volatility classification.

X. CONCLUSION

This study compares ten machine learning algorithms for binary market volatility classification, covering linear classifiers, support vector machines, instance-based learning, decision trees, and ensemble methods. The empirical results show that all ten algorithms achieve accuracy values above 80%, while the differences among the strongest-performing methods remain relatively small. Within the present experimental setting, these findings indicate that the current feature representation contains predictive information that can be utilized by several types of classification models.

Among the evaluated algorithms, Gradient Boosting records the highest overall accuracy and F1-score in this experiment. RBF SVM achieves the highest precision, indicating greater reliability among its high-volatility predictions, whereas Ridge Classifier records the highest recall and therefore detects the largest proportion of actual high-volatility observations. The results show that model selection involves different trade-offs rather than a single universally superior classifier. Applications in which missing a high-volatility period is particularly undesirable may place greater emphasis on recall, whereas settings that seek to limit unnecessary high-volatility warnings may assign greater importance to precision.

Another notable result is the comparatively small performance difference between the simple linear classifiers and the more complex ensemble models. The competitive results of Logistic Regression, Linear SVM, and Ridge Classifier suggest that the current feature representation can be utilized effectively by both simple and more complex decision functions. Moving averages, Bollinger Bands, volatility measures, momentum indicators, lagged returns, and other price-derived variables describe trend, short-term fluctuation, and temporal dependence. Since the study does not include an ablation over feature groups, the individual contribution of feature construction relative to the original market variables is not separately quantified here.

The study addresses future market uncertainty as a volatility classification task rather than as an estimation of exact future price changes. Volatility clustering introduces temporal structure into the target variable, allowing machine learning classifiers to distinguish between relatively low- and high-volatility conditions using historical information. Within the scope of the present experiment, this formulation produces consistent classification performance across several different model families.

From an application perspective, volatility classification may provide useful information for financial decision-support

systems. Predictions of high-volatility conditions can inform adjustments to risk exposure, position sizing, or other investment-related decisions. The present study does not directly evaluate such downstream strategies, but the classification results provide a basis for considering volatility-state predictions as an input to broader financial decision-making processes.

The findings should nevertheless be interpreted in light of several limitations. The analysis relies on historical price variables and technical indicators and does not incorporate macroeconomic information, geopolitical developments, or textual sentiment data. In addition, the rolling-volatility target contains temporal dependence that may contribute to the observed predictability. These factors restrict the extent to which the current results can be generalized to other forecasting environments.

The study also evaluates model performance using classification metrics rather than within a complete trading framework. Transaction costs, liquidity, slippage, and execution delays are therefore not reflected in the reported results. High predictive accuracy consequently should not be interpreted as direct evidence of trading profitability.

In summary, under the experimental setting examined here, market volatility classes can be identified with comparable effectiveness by a diverse collection of machine learning algorithms. Gradient Boosting records the highest accuracy and F1-score, RBF SVM the highest precision, and Ridge Classifier the highest recall, while the narrow performance range across the ten classifiers indicates that model choice should be guided by the evaluation measure most relevant to the intended application rather than by accuracy alone. These conclusions are bounded by the dataset, the volatility target definition, and the experimental configuration considered in this study; the section on future work outlines the extensions that would be needed to test whether they hold more broadly.

XI. FUTURE WORK

Future research can extend the present study by incorporating a broader range of explanatory variables. Macroeconomic information, including interest rates, inflation, employment statistics, and central bank policy indicators, could provide additional context for changes in market volatility. Textual information obtained from financial news, earnings-call transcripts, and social-media content could also be analyzed through sentiment-based features to capture market reactions that are not directly represented by historical price observations. Structured corporate information, including earnings reports, balance-sheet variables, and revenue growth, represents another potential source of explanatory information. Cross-market variables derived from market indices, commodities, bonds, and foreign-exchange markets could further be considered to capture relationships among different financial markets.

A second extension is to move beyond the current binary definition of volatility. Instead of separating observations only into low- and high-volatility classes, future experiments could define additional categories such as low, moderate, high, and extreme volatility. A multi-class formulation would provide a

more detailed description of market conditions. Alternatively, future volatility could be modeled directly as a continuous variable through regression, allowing the models to estimate volatility magnitude rather than assigning each observation to a discrete class.

Two further directions follow directly from the limitations of the present design. The first concerns the feature representation: an ablation study that removes individual feature groups, such as technical indicators, moving averages, lagged returns, and volatility measures, would make it possible to quantify how much each group contributes to classification performance and to separate the effect of feature construction from that of the original market variables. The second concerns the evaluation baseline: complementing the comparison among machine learning classifiers with simple persistence-based rules and with traditional econometric volatility models would indicate how much predictive value the learning-based approaches add beyond established benchmarks.

Future studies could also compare the traditional machine learning algorithms considered in this work with deep learning architectures designed specifically for sequential data. Long Short-Term Memory (LSTM) networks have demonstrated strong performance in realized volatility forecasting and in cross-sectional financial market prediction [35], and hybrid designs that ensemble such networks with econometric models have also been explored. Gated Recurrent Units and related recurrent variants have been examined for cryptocurrency volatility, while Transformer-based time-series models have been investigated for multi-market volatility prediction [36] and for exploiting intraday commonality. These architectures are capable of representing temporal dependencies and non-linear sequential relationships, and their performance could be compared directly with simpler machine learning approaches to determine whether their additional computational and structural complexity leads to meaningful predictive improvements.

Another important direction is to evaluate volatility predictions within practical financial systems. Predicted volatility states could be incorporated into trading strategies, portfolio optimization procedures, or risk-management frameworks. For example, forecasts could be used to modify position sizes, adjust stop-loss thresholds, or change asset allocations according to expected market conditions. Such experiments should also account for realistic constraints, including transaction costs, slippage, liquidity limitations, and execution delays, in order to determine whether improvements in predictive performance translate into measurable financial benefits.

Further investigation is also needed to examine robustness under changing market regimes. Financial markets can transition among bull markets, bear markets, crisis periods, and other states with substantially different volatility characteristics. As noted in Section IX, the results reported here rest on a single chronological partition of the data, so that the stability of the observed ranking across different time periods has not been examined. Evaluating the same classifiers over successive or rolling time windows, and applying approaches such as online learning, rolling model retraining, and regime-switching methods, would therefore indicate whether the models adapt when the underlying market environment changes.

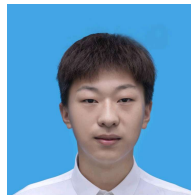
Finally, future research could examine model interpretability and explainability in greater detail. Feature-attribution and other model-agnostic explanation methods could be applied to investigate the variables that contribute most strongly to low- and high-volatility predictions, and to examine the internal behaviour of hybrid forecasting models whose individual components are difficult to interpret. A clearer understanding of individual model decisions would improve transparency and could make machine learning-based volatility classification more suitable for financial applications in which accountability and interpretability are important.

ACKNOWLEDGMENT

The authors would like to thank Tristan for his valuable contributions to this work.

REFERENCES

- [1] J. Wang, M. Zhou, X. Guo, L. Qi, and X. Wang, "A Markov regime switching model for asset pricing and ambiguity measurement of stock market," *Neurocomputing*, vol. 435, pp. 283–294, 2021.
- [2] E. S. Gunnarsson, H. R. Isern, A. Kaloudis, M. Ristad, B. Vigdel, and S. Westgaard, "Prediction of realized volatility and implied volatility indices using AI and machine learning: A review," *Int. Rev. Financ. Anal.*, vol. 93, p. 103221, 2024.
- [3] K. Christensen, M. Siggaard, and B. Veliyev, "A machine learning approach to volatility forecasting," *arXiv preprint arXiv:2601.13014*, 2026.
- [4] J. Gu, Z. Guo, J. Wang, L. Qi, S. Qin, and S. Zhang, "Optimization of multi-factory remanufacturing processes with shared transportation resources using the ALNS algorithm," *Int. J. Artif. Intell. Green Manuf.*, vol. 1, no. 1, 2025.
- [5] Y. Bai and C. X. Cai, "Predicting VIX with adaptive machine learning," *Quant. Finance*, vol. 24, no. 12, pp. 1857–1873, 2024.
- [6] S. Kumar, A. Rao, and M. Dhochak, "Hybrid ML models for volatility prediction in financial risk management," *Int. Rev. Econ. Finance*, vol. 98, p. 103915, 2025.
- [7] J. D. Díaz, E. Hansen, and G. Cabrera, "Machine-learning stock market volatility: Predictability, drivers, and economic value," *Int. Rev. Financ. Anal.*, vol. 94, p. 103286, 2024.
- [8] B. Han, A. Liu, J. Chen, and W. Knottenbelt, "Can machine learning models better volatility forecasting? A combined method," *Eur. J. Finance*, vol. 32, no. 13, pp. 1543–1564, 2025.
- [9] Q. Zhang, Y. Xing, C. Zhang, X. Sun, B. Hu, and A. Das, "Column generation algorithms for two-dimensional cutting problem with surface defects," *Int. J. Artif. Intell. Green Manuf.*, vol. 1, no. 2, 2025.
- [10] H. Zhang, D. T. Pham, and Q. Kang, "Improved fruit fly algorithm for multi-objective disassembly line balancing problem considering learning effect," *Int. J. Artif. Intell. Green Manuf.*, vol. 1, no. 2, 2025.
- [11] C. Zhang, Y. Zhang, M. Cucuringu, and Z. Qian, "Volatility forecasting with machine learning and intraday commonality," *J. Financ. Econometrics*, vol. 22, no. 2, pp. 492–530, 2024.
- [12] Y. Zhou, Q. Yan, X. Wu, Y. Bo et al., "DART-GNN: A dynamic recurrent graph neural network for multivariate time series anomaly detection," *Int. J. Artif. Intell. Green Manuf.*, vol. 1, no. 3, 2025.
- [13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [14] S. M. Daniali, S. E. Barykin, I. V. Kapustina, F. Mohammadbeigi Khortabi, S. M. Sergeev, O. V. Kalinina et al., "Predicting volatility index according to technical index and economic indicators on the basis of deep learning algorithm," *Sustainability*, vol. 13, no. 24, p. 14011, 2021.
- [15] J. Hosker, S. Djurdjevic, H. Nguyen, and R. Slater, "Improving VIX futures forecasts using machine learning methods," *SMU Data Sci. Rev.*, vol. 1, no. 4, p. 6, 2018. [Online]. Available: <https://scholar.smu.edu/datasciencereview/vol1/iss4/6>
- [16] S. H. Tang, M. Rosenbaum, and C. Zhou, "Forecasting volatility with machine learning and rough volatility: Example from the crypto-winter," *arXiv preprint arXiv:2311.04727*, 2023.
- [17] N. Roszyk and R. Ślepaczuk, "The hybrid forecast of S&P 500 volatility ensembled from VIX, GARCH and LSTM models," *arXiv preprint arXiv:2407.16780*, 2024.
- [18] E. Ramos-Pérez, P. J. Alonso-González, and J. J. Núñez-Velázquez, "Forecasting volatility with a stacked model based on a hybridized artificial neural network," *arXiv preprint arXiv:2006.16383*, 2020.
- [19] S. Yan, H. Guo, and S. Liu, "A partition stacking classification framework with oversampling for quality prediction of aluminum alloy ingots," *Int. J. Artif. Intell. Green Manuf.*, vol. 1, no. 1, 2025.
- [20] E. Rahimikia, S. Zohren, and S.-H. Poon, "Realised volatility forecasting: Machine learning via financial word embedding," *arXiv preprint arXiv:2108.00480*, 2021.
- [21] W.-B. Wang and Y. Wu, "Forecasting and hedging the volatility index of financial markets via a robust machine learning approach," *SAGE Open*, vol. 15, no. 4, p. 21582440251396044, 2025.
- [22] D. Somani, R. Gupta, S. Karmakar, and V. Plakandaras, "Supply bottlenecks and machine learning forecasting of international stock market volatility," *Finance Res. Lett.*, vol. 86, p. 108931, 2025.
- [23] Y. Fu, Z. Zhang, M. Huang, X. Guo, and L. Qi, "Multi-objective integrated energy-efficient scheduling of distributed flexible job shop and vehicle routing by knowledge-and-learning-based hyper-heuristics," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 9, no. 3, pp. 2137–2150, 2025.
- [24] X. Guo, F. Guo, L. Qi, J. Wang, S. Liu, S. Qin et al., "Modeling and optimization of multiproduct human-robot collaborative hybrid disassembly line balancing with resource sharing," *IEEE Trans. Comput. Social Syst.*, vol. 12, no. 5, pp. 2848–2863, 2025.
- [25] J. Wang, G. Xi, X. Guo, S. Liu, S. Qin, and H. Han, "Reinforcement learning for hybrid disassembly line balancing problems," *Neurocomputing*, vol. 569, p. 127145, 2024.
- [26] Kaggle, "Market trend analysis dataset," [Online]. Available: <https://www.kaggle.com/datasets/kundanbedmutha/market-trend-analysis-dataset>, 2024.
- [27] D. Chun, H. Cho, and D. Ryu, "Volatility forecasting and volatility-timing strategies: A machine learning approach," *Res. Int. Bus. Finance*, vol. 75, p. 102723, 2025.
- [28] T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *J. Econometrics*, vol. 31, no. 3, pp. 307–327, 1986.
- [29] R. Cont, "Empirical properties of asset returns: stylized facts and statistical issues," *Quant. Finance*, vol. 1, no. 2, pp. 223–236, 2001.
- [30] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [31] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [32] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*. ACM, 2016, pp. 785–794.
- [33] B. Yu, H. Du, and D. Zhang, "Forecasting the realized volatility of oil futures: Machine learning with cross-market spillovers," *Int. Rev. Econ. Finance*, vol. 108, p. 105176, 2026.
- [34] G. Dudek, P. Fiszeder, P. Kobus, and W. Orzeszko, "Forecasting cryptocurrencies volatility using statistical and machine learning methods: A comparative study," *Appl. Soft Comput.*, vol. 151, p. 111132, 2024.
- [35] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 654–669, 2018.
- [36] A. Li, "Volatility forecasting in global financial markets using TimeMixer," *arXiv preprint arXiv:2410.09062*, 2024.



Minghui Shi received his B.S. degree in Computer Science and Technology from Linyi University, Linyi, Shandong, China, in 2025. He is currently pursuing the M.S. degree in Artificial Intelligence with the School of Artificial Intelligence and Software, Liaoning Petrochemical University, Fushun, Liaoning, China.