DOI: 10.54254/2753-8818/2026.DL33364

Predicting Stock ETF Return Direction Based on Multi-Classification Models

Kunyuan Song

School of Statistics and Data Science, Southwestern University of Finance and Economics, Chengdu, China

sky270033@outlook.com

Abstract. In the world of quantitative finance, pinpointing the directional drift of asset returns often proves more reliable than chasing exact price points, a reality that has positioned trend forecasting as a cornerstone of modern trading strategies. This study moves beyond simple observation to build a comprehensive evaluative system focused on the Guotai Nasdaq 100 ETF, where four distinct analytical heavyweights—Logistic Regression, Random Forest, LightGBM, and XGBoost—are pitted against each other in a controlled head-to-head comparison. To maintain the sanctity of predictive signals, a feature space was meticulously woven together from simulated execution logs and external market variables while a zero-tolerance policy was enforced for any look-ahead bias that might skew the results. A deep dive into the performance metrics indicates that Random Forest consistently held the high ground, particularly when measuring overall accuracy and the harmonic balance of precision and recall, a success that essentially validates how ensemble-based bagging frameworks excel. While it was noticed that Logistic Regression still holds its own when it comes to flagging downside risks, the more complex boosting models seemed to hit a performance ceiling, likely because they were starved for the massive sample sizes they usually crave. By bridging this gap, this work offers a gritty, empirical look at model selection in noisy environments, highlighting that the real secret to success lies in matching the complexity of the algorithm with the actual statistical pulse of the market being traded.

Keywords: Exchange traded funds, Return Direction Prediction, Machine Learning, Random Forest, Multi-Classification Models

1. Introduction

The current state of financial market data is nothing short of a high-dimensional maze characterized by staggering levels of nonlinearity and persistent stochastic noise. If stock returns are viewed as the ultimate market barometer, it must be acknowledged they are pulled in countless directions by the constant friction between macroeconomic shifts, institutional capital flows, and the often-irrational whims of investor sentiment. This perfect storm of complexity is exactly why traditional linear models so frequently miss the mark, as they simply aren't built to digest the deep-seated interactions buried within the data.

DOI: 10.54254/2753-8818/2026.DL33364

Exchange traded funds (ETFs) stand out here because they offer a much steadier reflection of broad market trends compared to the erratic behavior of individual stocks, which are far too prone to idiosyncratic shocks. For anyone sitting at a trading desk, the question of whether a return is moving up or down usually carries more weight than knowing the exact decimal point of that movement, so reconfiguring this as a classification task is not just a statistical choice but a practical necessity that makes the results far more robust. The tools of the trade have been seen to evolve from the old-school ARIMA and GARCH models into the modern machine learning era where Support Vector Machines once led the charge [1].

Today, Random Forest has earned its reputation for being remarkably tough in high-noise environments [2], while the latest gradient boosting frameworks like XGBoost and LightGBM have pushed the boundaries of efficiency in everything from trend spotting to credit checks [3,4]. Despite these leaps forward, a strange lack of research was noticed that pulls together granular transaction data with wider market signals specifically for ETF forecasting [5], so this study steps into that void to see which of these modern algorithms actually delivers the best predictive edge when the stakes are real.

2. Data preprocessing

The raw material comes from a blend of historical market deep-dives and carefully crafted scenario simulations, starting with a reconstruction of daily market conditions from mid-2013 through late 2025. By applying a set of rigorous trading rules, a baseline of 411 transaction records was generated that capture the essential pulse of the market, including price swings, volume spikes, and cumulative returns under a straightforward full-buy-or-sell logic starting with a million yuan. To ensure the models had enough substance to learn from, a resampling method based on historical distributions was used to expand the sample size, essentially perturbing the parameters to create extra training data that still feels and acts like the real market.

Once this raw data was in hand, the process of standardizing the mess began by pairing up buy and sell records chronologically to map out complete trading cycles. The filters used were brutal, tossing out any trade that lasted less than a day to avoid getting bogged down in noise while minorizing any extreme return values that might distort the view. This left 192 rock-solid transactions split between upward and downward moves, where the target variable was defined simply: if a sell return was positive, it was a win. It is realized that this binary approach glosses over the fact that a tiny gain and a massive rally reflect different market moods, but it matches the binary nature of most trading decisions where the first question is always about direction. To keep the predictions honest, the feature set was built using only what was known at the moment of purchase, covering momentum through various return windows, risk through volatility measures, and short-term strength through indicators like the RSI. This entire pipeline of cleaning, pairing, and scenario-driven expansion was designed to produce a modeling dataset that is both clean enough for a computer to read and rich enough to reflect actual market dynamics.

3. Research methods and model construction

3.1. Modeling approach for return direction classification

Instead of chasing exact return magnitudes in the high-entropy Nasdaq 100 space, this study finds that pivoting toward directional drift creates a much sturdier analytical floor. This strategic shift helps filter out the erratic and structural-less flicker of short-term price action that so often crushes

DOI: 10.54254/2753-8818/2026.DL33364

brittle models relying on exact price points. The entire prediction task was reconfigured into a simple win-loss scenario by mapping specific market micro-states captured at the entry threshold directly to their eventual success. To keep the training process resilient across diverse regimes, the modeling pipeline uses an expanded and scenario-driven dataset while the underlying probability distributions are strictly guarded. The experimental design rejects any reliance on a single estimator; instead, a comparative gauntlet was built where every learner, from linear baselines to adaptive ensembles, is tested against identical data splits to ensure that performance gaps reflect model logic rather than some random artifact or luck of the draw.

3.2. Research methods

To do a systematic comparison between various algorithms being used in the ETF return direction prediction task, this study adheres to a comparative approach of simple to complex, as well as linear to non-linear, and systematically introduces four classification models, including Logistic Regression, Random Forest, XGBoost and LightGBM. Each of the models forecasts the direction of the returns at the time of sell using market state characteristics between the time of purchase and the time of sale.

The Logistic Regression is a standard generalized linear model which develops the relationship between variables and the likelihood of an occurrence based on the logit formulated as follows:

$$P(Y=1)=\frac{1}{1+e^{-(\beta_0+\beta_1x_1+\beta_2x_2+\ldots+\beta_nx_n)}} \tag{1}$$

It has a concise model form and does not have a volatile parameter estimation process; the model is also easily interpretable. It is used as a benchmark to determine the existence of a market signal structure that is linearly separable. The model gives the probability of a transaction to give a positive payoff and classification is selected by determining a threshold [6].

Random forest is an ensemble learning algorithm which uses the Bagging approach. It classifies by building decision trees with multiples of predictions, which are aggregated and developed in the following way:

$$\hat{y}=mode\{h_1(x),h_2(x),\ldots,h_T(x)\} \tag{2}$$

In this model, randomness in sample and feature dimensions is introduced, thus minimizing the variance of the modelling as well as overfitting. It automatically captures nonlinear interactions of high order between features and usually has a high robustness in small order nonlinear financial data [7].

Both XGBoost and LightGBM belong to a framework known as Gradient Boosting Decision Tree (GBDT), which is carried out efficiently. XGBoost is more effective in boosting model generalization with regularization terms on the objective function and information on the second-order gradient used in optimizing the tree structure. Its conception is as follows:

$$\hat{y}_i=\sum_{k=1}^{K}f_k(x_i) \tag{3}$$

The objective function is given by

$$Obj=\sum_{i=1}^{n}L(y_i,\hat{y}_i)+\sum_{k=1}^{K}\Omega(f_k), \tag{4}$$

DOI: 10.54254/2753-8818/2026.DL33364

while the regularization term is given by

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda\|w\|^2 \tag{5}$$

LightGBM, in its turn, uses a decision tree algorithm which relies on a histogram and a leaf-wise growth strategy which improves the training efficiency, particularly with large data sets significantly. Its fundamental inventions are Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB).

These two models, through iteration and gradual residual, block out the complex nonlinear functions. The new advanced Boosting models proposed, in turn, will test the hypothesis that, given sufficient sparse conditions of the samples, such convoluted sequential structures of learning can provide better performance improvements than Random Forest.

Each model is also trained and evaluated on the same dataset split and feature set in order to make the comparison fair. In terms of evaluation measures, this paper also presents precision, F1-score, and accuracy. The discriminatory values of the models of the upward and downward classes are determined separately so as to better represent the use value of each model in the investment decision making backgrounds.

4. Results analysis and discussion

4.1. Descriptive statistics

Descriptive statistics of the core feature variables before model training were computed to gain understanding of the fundamentals of data distribution. Table 1 indicates that the standard deviation of the closing price is 1.610 with the minimum of 0.979 and the highest of 5.525 with an overall increase of more than 5.7 times of the close price of this ETF within the sample period. The average daily change of 0.03 was not significantly different, so the long-term returns are in line with the Efficient Market Hypothesis, but the standard deviation was large (1.80) which suggests that prices vary massively on a daily basis. The mean volume of trading was 5.8 million shares although it had a high standard deviation of 14.11 million shares. The fact that order-of-magnitude difference between maximum and median is large indicates highly skewed distribution of market activity, whereby there are extreme volume expansions. Average volatility of 20 days was 1.40% at a standard deviation of 0.30% and it showed that the overall volatility levels are fairly constant. The average RSI stood at 52.65, with the range of 12.82 and 85.05 as the lowest and highest respectively, which did not comply with the traditional indicators, hence indicating a time of highly exaggerated down and up oversold positions in the market. These statistical features give a preliminary understanding of the sophisticated nature of the market states and as such are intuitively explained as to why further model prediction proves to be very difficult.

Table 1. Descriptive statistics of main market feature variables

Feature Variable	Sample Size	Mean	Std Dev	Min	Max
Closing Price	2886	2.766	1.610	0.979	5.525
Daily Return (%)	2886	0.03	1.80	-4.70	4.70
Volume (million shares)	2886	5.80	14.11	0.05	424.01
20-Day Volatility (%)	2886	1.40	0.30	0.60	3.10
RSI (14-day)	2886	52.65	13.33	12.82	85.05

DOI: 10.54254/2753-8818/2026.DL33364

4.2. Overall comparison of model prediction performance

Just as in the case of the united dataset split, Table 2 gives the prediction performance of each of the models. Based on the overall accuracy, the highest value 0.552 was obtained with the help of the Random Forest model, which suggests that in the given conditions under which the data were represented, the ensemble learning mechanism of the model proves to be more effective in the description of the nonlinear patterns of ETF return direction. LightGBM performed with a 0.523 accuracy level, which is in the middle. Accuracies of Logistic Regression and XGBoost were 0.496 and 0.481 respectively, which are also lower than the accuracy of random forest.

Table 2. Model performance summary table

Model	Accuracy	Precision	Recall	F1-Score	Up Precision	Up Recall	Down Precision	Down Recall
Logistic Regression	0.496	0.507	0.496	0.476	0.532	0.306	0.481	0.705
Random Forest	0.552	0.550	0.552	0.546	0.561	0.664	0.538	0.430
LightGBM	0.523	0.522	0.523	0.522	0.542	0.572	0.500	0.470
XGBoost	0.481	0.480	0.481	0.478	0.466	0.401	0.493	0.558

The comparatively poor results of the XGBoost in this research are to be considered carefully. Although it can be partially linked to small sample size and the possibility of tendency to noise, it does not mean that there is another relational drawback of the model. Practically, XGBoost has been observed to be very sensitive to hyperparameter settings, such as tree depth, learning rate and strength of regularization. Hyperparameter tuning was not thoroughly studied because of this research topic. Perhaps even with a much more severe regularization or more conservative parameter choices, XGBoost would be more effective at regulating overfitting and would be able to attain better predictive performance.

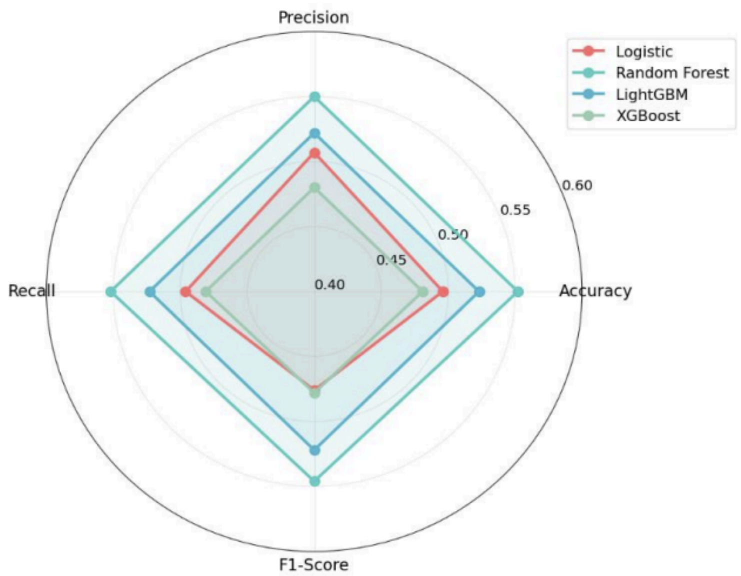

Figure 1. Radar chart of four model performances

Random Forest also enters into the first place in the overall F1-score outcome, which also proves that it is superior in terms of the stability and stability of prediction in the existing experimental environment. It is also notable that accuracy shows how well the model has been correct, the

DOI: 10.54254/2753-8818/2026.DL33364

percentage of all samples whose prediction is correct, whereas precision shows the dependability of samples with positive prediction, the percentage of all positive predictions which are also accurate. However, in the context of financial decision-making, accuracy and precision imply different practical implications; the former is simply the total judgment ability of the model, and the latter has to do more with the sort of success that one trades on around model signals can achieve.

To more intuitively compare the overall performance of the models, a radar chart was plotted in Figure 1. This chart illustrates how the four models perform on a single coordinate system on four dimensions (Accuracy, Precision, Recall, F1-score) where each axis in the system corresponds to a particular dimension and the score of the models on each dimension on the axis is linked up to create a closed polygon. As shown in Figure 1, it has been made evident that the Random Forest model has taken the highest area in the polygon, especially in comparison with other models, which shows a significant difference in aspect of their performance in terms of Recall, as well as, F1-score. This once again supports the fact that it is the leader in terms of overall predictive capability. Conversely, the Logistic Regression and XGBoost polygons are smaller and the performance is more or less balanced, but worse in terms of dimensions. LightGBM is in the middle having average results on all parameters. The multi-dimensional visual comparison will enable to better in comparison the strengths and weaknesses of each of the models, instead of only comparing single indicators.

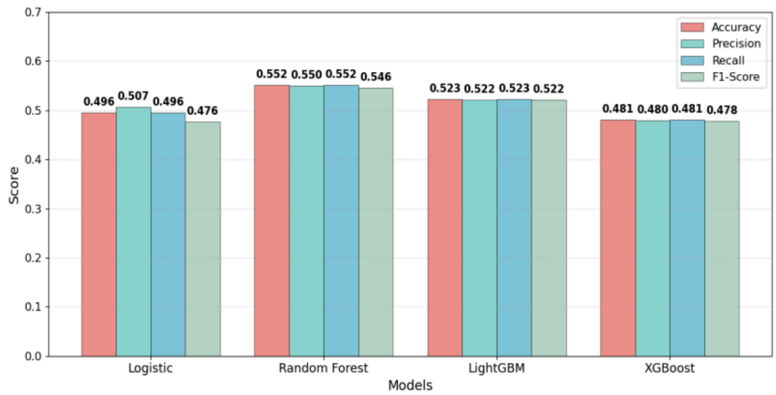

Figure 2. Model performance comparison bar chart

In an attempt to confirm the above results, a model performance comparison bar graph was plotted in figure 2. This chart shows the individual values of the four models on the four assessment measures in a bar chart format. Based on Figure 2, numerical differences are more visible: the leading role is preserved by Random Forest in all the four metrics with the largest differences being on the recall metric, where it scores much higher than Logistic Regression and XGBoost. The lightgbm is performing comparatively, as all the measures are within a range of 0.52. It is important to note that the precision of Logistic Regression is marginally greater than the recall whilst the reverse is observed with random forest (recall greater than precision) as they both vary in terms of the decision biases inherent in the models.

4.3. Analysis of classification capability by return direction

In order to understand the differences between the behavior of the models better, this paper further subsumes their precision and recall rates based on the upward/downward direction.

DOI: 10.54254/2753-8818/2026.DL33364

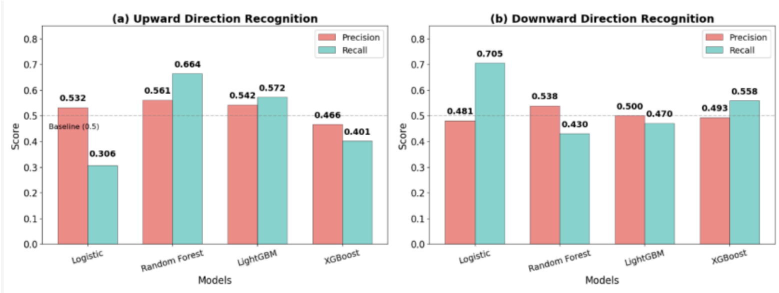

Figure 3. Bar Chart of Classification Performance Comparison by Up/Down Direction

In order to visualize these differences with a greater classicism, the bar chart of performance of classification by up/down direction was plotted in Figure 3. This chart has two subplots, respectively, the accuracy and the recall of each model to the classes upward and downward.

Based on Figure 3-a (identification skill on the direction facing up), the disparities among models are immense. The upward recall of the Random Forest is 0.664 which is high compared to all other models and this implies that it has a unique benefit in embracing upward opportunities though it comes at a slight cost by sacrificing precision. The upward recall of the Logistic Regression is the lowest positive meaning is that it is conservative i.e. it judges more on the downwards than the upwards.

Figure 3-b (recognition situation downwards direction) reflects the converse outcome entirely: the downward recall of the Logistic Regression models is the highest of all, which proves its high sensitivity to the downside risk. The downward recall of the random Forest is stable and implies that the random forest forecast is more inclined to the upwards direction. The performance of LightGBM is not too biased in any direction, and nobody can demonstrate the clear advantage of XGBoost. These scissors difference clearly shows that the intrinsic decision preferences of the various models are opposite: that when the sample conditions of this study, the linear model Logistic Regression is more conservative, and approaches risks; whilst the random forest is more aggressive, and approaches opportunity.

This disparity is not accidental as it has much to do with the algorithmic processes of every model and can be explained reasonably regarding the theoretical viewpoint. Examining the algorithmic approach, as the generalized linear regression model, which is called Logistic Regression, the boundary of its decision tree is built on the linear relation between the features and the log odds and the parameter weights are obtained by the maximum likelihood estimation. This mechanism identifies its sensitivity to changes in features and a bias toward conservative judgments at the feature distribution boundary, when the evidence is weak enough to justify an upward movement, the model would prefer to categorize it as downward and is a conservative property of the risk identification and loss avoidance process. Random Forest, in contrast, being an ensemble learning model, with the strategy of Bagging, is predictive by building multiple decision trees independently, and classifying the votes. All trees are trained with randomly selected samples and randomly chosen feature subsets and are capable of modeling complex nonlinear interactions between variables and have enhanced resistance to outliers and noise.

This mechanism of collective decision has the advantage that it is more prone to ensuring potential upward opportunities in situations of uncertainty or local information, and it has an aggressive orientation towards grabbing returns. Gradient boosting algorithms such as XGBoost and LightGBM instead embrace the concept of Boosting and independently fit the remainder of the past performance until the model gradually improves on the performance of the challenging samples. The continuous reinforcement of their discrimination property on boundary samples through this sequential learning process improved their discrimination capacity of boundary samples in training which, in this case, is more sophisticated discrimination between upward and downward classes and more balanced prediction performances.

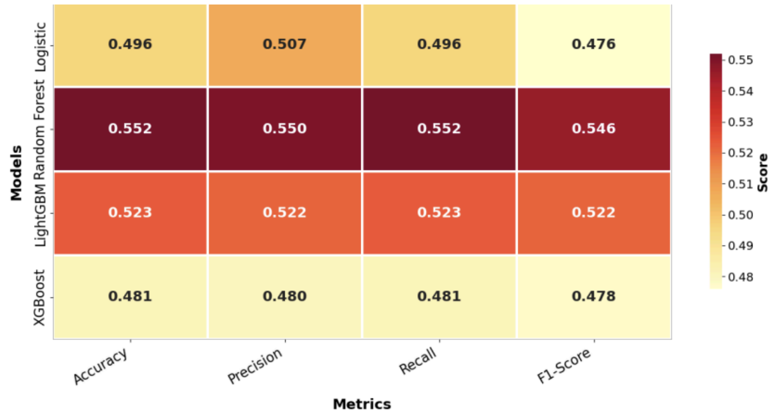

Figure 4. Model performance heatmap

More so, Figure 4 presents the heatmap that the author applies to fully demonstrate the performance of each model using various metrics. Darker colors in the heatmap mean that the scores are high. It is intuitively apparent that: The darkest colors are on all metrics in Random Forest which is doing the best; the darkest colors are on the downward recall, but the lightest on the upward recall in Logistic Regression, this is an evident bias; the lightest on all metrics in the XGBoost, with a relatively weaker performance. It is a fast way of pinpointing the strengths and the weaknesses of each of the models through this visualization technique.

The author summarizes the detailed performance indicators of each model in the Table 2 so as to enable an easy reference and comparison. This table explicitly indicates the particular scores of all models within four overall measures and eight disaggregated evaluations that allow using the table in further analysis and discussion to have data base.

4.4. Chapter summary

There are three major factors which give rise to model performance differences.

To begin with, insufficient information about features. Historical simulation transactions are the sources of the data, and the core variables are reduced to price, position, and returns. High-frequency technical indicators, market sense, and the lack of macroeconomic indicators make the models limited in the information that they can predict.

In order to see how Random Forest performs best, the top eight ranks of feature importance were analyzed (as shown in Figure 5) which shows that the most important features in training are the following, namely: 20-day Historical Volatility, and 5-day Cumulative P Return which is in line with the basic risk return trade-off, that is, the volatile reflects the market risk, and momentum reflects the short run returns. Even "RSI (14-day) is ranked high, and the importance of the short-term strength indicators are proven. On the other hand, the "Average Volume" is not significantly important and it is possible to infer that the volume does not predict much in relation to price in this sample. The results inform the future design of feature engineering to focus on volatility and momentum indicators, as well as demand more powerful price-volume gauges.

DOI: 10.54254/2753-8818/2026.DL33364

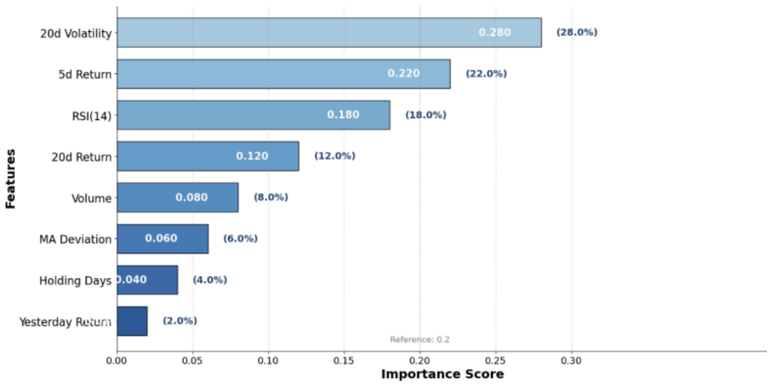

Figure 5. Random forest feature importance ranking chart

Second, problems in the structure and size of the samples. The historical transaction books had a record of 411 only. Statistical independence and representativeness are also a concern even following expansion. Small to medium samples are also well-constructed using tree-based models, hence the good performance of Random Forest. XGBoost and other gradient boosting models are more sensitive to noise, which explains why more sophisticated Boosting models never outperformed the simpler structurally, but still, random Forest.

The fact that random forest outperforms XGBoost looks like a point to be examined further. The main distinction between the two algorithms is the way they deal with noise and small samples, algorithmically. Random Forest uses Bagging: this is a technique that constructs parallel trees through bootstrap sampling and combines predictions by means of voting or averaging. The variance is decreased by this mechanism - even when each tree overfits noise, averaging outperforms it, resulting in more generalization. Having a limited sample of the valid transactions (and a small signal-to-noise part) this joint decision-making is valuable.

On the other hand, a Boosting algorithm such as XGBoost models are built in series and rely on previous prediction residues. This compels the model to concentrate on hard misclassified samples in the past. Boosting can exaggerate the noise in noisy data, making it appear to be signal, and overfitting the model. Biaswise, Boosting minimizes bias but might amplify variance quickly when operating on small samples. Random Forest, uses Bagging to increase the reduction in variance and works better in cases where the amount of data is restricted. The smallest accuracy (0.481) of XGBoost indicates its complicated design that is not very effective in a low-signal condition.

Third, natural uncertainty in financial speculation. The returns of ETF are vulnerable to external shocks and the short-term effect is a near random walk. No statistical formula based entirely on past prices and trading activity can have an unwavering edge. There was no model with higher accuracy than 0.60, which proves that short-term ETF direction is considerably random. In an environment where the features are purely historical in their derivation such that the predictability upper of the market is constrained- as weak-form market efficiency would theoretically imply [8].

A unified comparison reveals that on all accuracy, F1-score, and upside the determining parameters, and under the current circumstances, Random Forest is the most viable approach to choose. LightGBM is not very great, showing that it has the ability to be nonlinear. Logistic Regression is good at downside but not forecasts. XGBoost does not work well, suggesting that it relies on better features and larger datasets [9].

Importantly, the findings imply that there is a natural maximum direction prediction of ETFs when using transaction-derived data only. The importance analysis of features demonstrates that volatility and momentum are the main aspects to look at to inform future feature engineering. Multi-source data fusion, deep time-series models and real trading backtests will need to be advanced to develop useful investment value [10].

5. Conclusion

This paper systematically compares four models, namely: Logistic Regression, Random Forest, LightGBM and XGBoost, at the operational task of ETF return direction prediction based on historical simulated transaction data and their extended samples. Findings of the research show that in circumstances, where minimal information about features is available and the sample size is small, the general predictability of all models were not more than 0.60. This makes the theoretical knowledge of the weak-form efficiency of financial markets true, that the direction of short-term returns has a high level of uncertainty and randomness, and the inability to predict with good consistency high-precision predictions based on the variables derived using history transactions alone. Regarding the model comparison approach, the most accurate, F1-score, and upward direction recognition are observed between Random Forest, which demonstrates the strengths of the ensemble learning approach toward the capabilities of dealing with nonlinear financial data and small- to medium-size data samples. The analysis of feature importance also demonstrates that 20-day Historical Volatility, and 5-day Cumulative Return are some of the key variables in the making of predictions and these variables are in line with the basic financial reasoning of risk and return. The performance of LightGBM is average, which confirms the capacity of the gradient boosting framework to possess nonlinear structures. The strength of the Logistic Regression in generalization is less good; however, the downside identification is more beneficial than using Multi-ensional Statistics. The results of XGBoost were not expected, which indicated that this algorithm was more sensitive to the size of the data and the quality of features.

The inability of XGBoost to exhibit anticipated performance is fundamentally related to the inaccuracy in considering that the prediction algorithms inherent to XGBoost algorithm in workout under such conditions as a handful of fewer 192 small-sample transactions and a low signal-to-noise ratio XGBoost is prone to over-attending to noise, a phenomenon that results in overfitting. Contrastingly, the noise resistance of the parallel Bagging strategy by the Random Forest, also based on averaging variance, is higher. This implies that in finance prediction, a more complicated model should suit a larger amount of data--when the samples are few an ensemble strategy (Bagging) can be simpler than an iterative strategy (Boosting).

The research also has some limitations: data origin is mostly simulated transactions, which limits the authenticity of the sample and the scale; feature system addresses the variables of transaction origins, which does not include the multi-dimensional information processing, such as macro factors and sentiment; the research is at the level of statistical prediction and does not take into account the actual costs of trading strategies to test them back. Future studies can follow three directions further: first, provide real market data across longer periods in time, incorporating multi-source heterogeneous feature regimes in the form of macroeconomics, capital flows and investor sentiment to build a richer feature space; second, investigate deep time series models like LSTM and Transformer to gain the capacity to learn dynamic dependence structure of returns; third, embed prediction models into a full quantitative trading scheme, including slippage, transaction costs, and position management to test their value of investment in practice in back-testing. As a result of this study, an insight has been obtained by far, in terms of financial prediction: data quality, feature engineering and problem boundary definition are frequently far more important than simply seeking complexity of the algorithm. Strict experimental design and admiration of the randomness of the markets are the principles, under which one can get trustworthy conclusions.

DOI: 10.54254/2753-8818/2026.DL33364

References

[1] Zhang, Z., Zohren, S., & Roberts, S. (2023). Deep learning for portfolio optimization. The Journal of Financial Data Science, 5(1), 8–20.

[2] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.

[3] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794).

[4] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In Advances in Neural Information Processing Systems (Vol. 30).

[5] Tang, Y., Liu, Y., & Zhang, J. (2022). A survey on machine learning models for financial time series forecasting. Neurocomputing, 512, 1–17.

[6] Piovezan, R. P. B., Silva, E. C., & Pinto, A. C. F. (2024). Machine learning method for return direction forecast of exchange traded funds (ETFs) using classification and regression models. Computational Economics, 63(5), 1–25.

[7] Ni, W., Tian, M., Jiao, Y., & et al. (2025). Multi-model gold stock price prediction based on machine learning. Changjiang Information & Communications, 38(2), 169-172.

[8] Gurrib, I., & Kamalov, F. (2021). High frequency return and risk patterns in U.S. sector ETFs during COVID-19. SSRN Electronic Journal.

[9] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), 654–669.

[10] Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review. Applied Soft Computing, 90, 106181.