How well do machine learning models in finance work?

Yeonchan Kang\(^a\), Doojin Ryu\(^a,^*\), Robert I. Webb\(^b\)

\(^a\) Department of Economics, Sungkyunkwan University, Seoul, Korea
\(^b\) McIntire School of Commerce, University of Virginia, Charlottesville, VA, USA

*Corresponding Author. E-mail: doojin.ryu@gmail.com

ORCID

Doojin Ryu (doojin.ryu@gmail.com): https://orcid.org/0000-0002-0059-4887
Robert I. Webb (riw4j@virginia.edu): https://orcid.org/0000-0003-1714-5778

Highlights

- Tree-based machine learning models exhibit relatively superior performance in stock return prediction.
- 36-month stock return momentum is a key variable for the machine learning prediction.
- Interpretable machine learning approaches are used to understand the relationship between predictors and predictions.

Abstract

This study evaluates the prediction power of machine learning models and identifies key variables for the stock return prediction. By analyzing various firm characteristics and macroeconomic variables for a recent dataset, we find that tree-based models exhibit relatively superior performance. The permutation feature importance method yields that the 36-month stock return momentum is the key variable for the prediction in the Korean equity market, a leading emerging market. The importance of the key variable is also elucidated using partial dependence, individual conditional expectation, and accumulated local effect plots.

Keywords: Emerging market; Feature importance; Interpretable machine learning; Stock market prediction

JEL classification codes: C45(Neural Networks and Related Topics), C53(Model Evaluation, Validation, and Selection), C55(Large Data Sets: Modeling and Analysis), G0(General), G11(Portfolio Choice, Investment Decisions), G17(Financial Forecasting and Simulation)

1. Introduction

Whether the stock market dynamics can be predicted is an intriguing question, and according to the weak-form or semi-strong-form efficient market hypotheses, future stock returns cannot be exactly predicted without inside information (Fama, 1970). The ongoing debate on this issue is supported by a large body of empirical asset pricing literature with different views. Israel, Kelly, and Moskowitz (2020) argue that return predictability exists even in a theoretically efficient market. The rise of machine learning, which has demonstrated remarkable out-of-sample prediction performance is changing entire industries. Recent interdisciplinary studies show that machine learning methods are effective in explaining asset price dynamics, economic forecasting, portfolio management, and return prediction through machine learning methods.
Though the machine-learning methods effectively detect non-linearity, consider complexity, and identify potential predictors, their black-box nature often results in the lack of economic implications and interpretations. Further, while most machine-learning approaches in this field are adopted for developed markets such as the US and European markets or huge markets such as the Chinese market, studies on emerging markets and small economies are relatively scarce.

Our study is motivated by the following limitations of machine-learning applications in financial studies: 

i) For a wider and more universal adoption of machine learning in economic research, one should be able to answer why and how the results are obtained, and

ii) Despite the growth of machine learning approaches in economic research and practice, there is still a need for broader applications in various financial markets. In this study, we explore the various machine learning models for stock return predictions in the Korean market, a representative and leading emerging market. Importantly, we try to identify the key variable using interpretable machine learning methods which make a user understand the reason for the decision of the machine. We assess the predictive performance of models using the recent dataset and employ interpretability tools such as permutation feature importance (PFI), partial dependence plots (PDP), individual conditional expectations (ICE), and accumulated local effects (ALE). These methods provide insight into how various firm characteristics and macroeconomic variables play roles in stock market predictions under the machine-learning framework.

Financial machine learning is at the intersection of several fields, including engineering economics, financial economics, statistics, computer science, and applied mathematics. By depending on a bigger and higher-dimensional dataset, machine learning models tend to be robust to multicollinearity and endogeneity issues compared to traditional econometric models. Following the seminal paper of Gu, Kelly, and Xiu (2020), who demonstrate the superior out-of-sample predictive performance of machine learning models using various firm characteristics and macroeconomic variables for the US stock market, subsequent studies apply machine learning approaches to different financial markets and asset classes. For example, Leippold, Wang, and Zhou (2022) adapt machine learning models to the Chinese stock market, accounting for its unique market characteristics such as private investor dominance, government-controlled financial systems, and a limited history of short selling. Similarly, Bali, Goyal, Huang, Jiang, et al. (2020) use advanced machine learning models, including long short-term memory, to predict US corporate bond yields. Bianchi, Büchner, and Tamoni (2021) show that extreme tree and neural network models can detect bond yield fluctuations in the US Treasury market for large samples. Studies on other worldwide markets demonstrate the

---

For the characteristics of the Korean equity market and economy and its influence as a leading emerging market, refer to the recent empirical studies (e.g., Ahn and Ryu, 2024; Ham, Ryu, and Webb, 2022; Ham, Ryu, Webb, and Yu 2023; Kim, Ryu, and Yang, 2019, 2021; Kim and Ryu, 2020; Lee, Ryu, and Yang, 2021; Seock, Cho, and Ryu, 2021, 2022; Ryu, Yang, and Yu, 2022; Kim, Ryu, and Yu, 2022; Chung, Jhang, and Ryu, 2023; Park and Ryu, 2023; Ryu, Webb, and Yu, 2024).
potential of machine learning models in enhancing the prediction for individual firm returns and effective portfolio construction (Lalwani and Meshram, 2022; Marsi, 2023; Rubesam, 2022). Unfortunately, a comprehensive understanding of the interpretability of machine learning is lacking in previous studies. To overcome this, we employ machine learning interpretability methods to identify the variables that significantly influence stock return predictions.

Estimating the importance of variables is an important tool for the integration of machine learning within finance research (López de Prado, 2018). The black-box problem of machine learning models makes it difficult to determine which variables are the most important and interpret the relationships between predictors and predictions. Researchers propose different methodologies to address this issue. For instance, Friedman (2001) introduces PDP to gain insight into model predictions, thus effectively exploring the relationship between a target variable and its predictors.

Considering that the PDP ignores the correlation between predictors, Goldstein, Kapelner, Bleich, and Pitkin (2015) introduce ICE plots showing how predictions for individual instances change with feature variation, providing more individual-level insights compared to the aggregate approach of PDP. Apley and Zhu (2020) propose ALE as an extension of PDP, which is effective for analyzing datasets with extensive predictor interaction. Especially for financial data with a low signal-to-noise ratio, machine learning interpretation is more important. Research also applies these methods to explain the prediction of various financial assets as follows. Leung, Lohre, Mischlich, Shea, et al. (2021) investigate the key features of predicting stock returns using PDP. Liang and Cai (2022) use models such as long short-term memory and convolutional neural networks to price European options and ALE to identify the influential factors in the predictions.

Our study identifies the most effective machine learning model for stock return prediction on the Korean market and determines which firm characteristics and macroeconomic variables have the most significant impact on stock return prediction under the machine learning framework. We construct a comprehensive high-dimensional dataset based on the numerous variables validated in previous research. It is a rare case that a specific machine learning model outperforms and dominates all other models in all datasets. Therefore, this study compares various machine learning models and conducts a horse race for the models to identify the most effective ones for predicting stock returns. Linear machine learning models, such as principal component regression (PCR), partial least squares (PLS), elastic net (ENet), tree-based models such as random forest (RF), extreme gradient boosting (XGB), light gradient boost machine (LGBM), and the most complex model neural network (NN) are used in this study. Key variables are then identified with stock returns analyzed using PFI. We also use other visualization approaches such as PDP, ICE, and ALE to interpret the interaction between key variables and stock return prediction.

Our empirical results indicate that non-linear models, particularly tree-based models exhibit superior performance in stock return prediction. However, (NN) model, underperforms compared to tree-based models, which might be related to the data limitations in emerging markets with relatively immature status, short history, and insufficient data observations. Our results and properties of the dataset somewhat contrast with extant studies such as Gu, Kelly,
and Xiu (2020), who use data going back to the 1950s, a depth of data often unavailable in emerging markets. Another finding is the importance of momentum-related variables. The 36-month momentum, defined as the cumulative returns from 36 to 13 months prior to each period is the key variable to predict stock return via machine learning. The visualization through interpretable machine-learning methods confirms the importance of the 36-month momentum for stock return prediction and improves our understanding of the machine-learning mechanism.

The remainder of this paper is organized as follows. Section 2 describes the construction process of the dataset. Section 3 explains the machine learning models used and the various methods employed for interpreting these models. Section 4 discusses the empirical results and their implications. Finally, Section 5 concludes the study with a summary of the findings and the related insights.

2. Data

Our dataset consists of data from 2005 to 2022 on all (3,616) firms listed on the Korea Exchange (KRX) and includes 80 firm characteristics and 10 macroeconomic variables. The target variable for prediction is the monthly excess return of each stock. Given the lack of reliable statistical data on the Korean market for periods before 2005, the analysis period is set from 2005 to 2022. Data were collected from DataGuide and the Economic Statistics System (ECOS) of the Bank of Korea.

The firm characteristics are adapted from Green, Hand, and Zhang (2017), as they are widely used in financial research, and are tailored to the Korean context. Owing to the differences in accounting standards and reporting periods between the US and Korea, about one-third variables are omitted. As a result, 64 of the 94 characteristics are converted to Korean data. Additional 16 characteristics (capxint, ef, cfroa, chadv, chato, chpm, grgw, indsale, pchcapx, rdbias, roa, roe, sacc, securedind, xadint, and xrdint) are included to increase the size of the dataset, given their positive correlation with the performance of machine learning models. These additional characteristics include percentages, industry sectors, and growth rates to provide more information about the Korean market. The final sample comprises nine monthly and 71 annual firm characteristics. For industry classification, the Korean Standard Industrial Classification (KSIC) intermediate classification, which is similar to the US Standard Industrial Classification (SIC) digit code, was used. The 77 categories of the KSIC closely match the 74

---

2 The Korean stock market consists of three exchanges: the Korea Composite Stock Price Index (KOSPI), the Korea Securities Dealers Automated Quotations (KOSDAQ), and the Korea New Exchange (KONEX). The KOSPI mainly lists large and medium-sized companies, KOSDAQ lists small and medium-sized companies and start-ups, and the KONEX facilitates the trading of newly established companies. We analyze all stocks traded on these three exchanges from 2005 to 2022.

3 The 30 characteristics (aeavol, barspread, betasq, chfeps, chnanalyst, cinvest, convind, disp, ear, fgr5yr, idiovol, ill, IPO, lev, maxret, ms, nanalyst, nincr, pricedelay, ps, rd_mve, roavol, sfe, std_dolvol, std_turn, stdacc, stdef, sue, tb, turn) are excluded due to the limitation mentioned in the data section.
codes of the SIC, ensuring the comparability and relevance of the industrial classification for analysis. Table 1 provides the description and sources of the firm characteristics used in this study. For clarity and ease of understanding, the description of these data aligns with the acronyms used in CRSP/Compustat. This facilitates the identification and comparison of firm characteristics by making the data more accessible to those familiar with these widely used databases.

[Table 1 inserted about here]

In addition to firm characteristics, macroeconomic variables also play an important role in influencing stock returns. These variables typically include government-published data, stock market indices, and bond interest rates. This study adopts the macroeconomic variables proposed by Welch and Goyal (2008), which consist of the following eight variables: dividend price ratio ($dp$), treasury bill rate ($tbl$), earnings price ratio ($ep_y$), book-to-market ratio ($bm_y$), term spread ($tms$), long-term yield ($lty$), default yield ($dfy$), and inflation ($infl$). In addition, two variables are included to reflect market sentiment: the Economic Sentiment Index (ESI) and the News Sentiment Index (NSI). The ESI, provided by the Economic Statistics System of the Bank of Korea, combines the Business Survey Index (BSI) and the Consumer Survey Index (CSI) and is calculated as a weighted average of these indices. The NSI is generated through web crawling and natural language processing and provides a quick reflection of market sentiment without direct surveys or text data processing. All macroeconomic variables have a monthly frequency.

The comprehensive dataset created for this study includes 71 annual frequency firm characteristics, nine monthly firm characteristics, and 10 monthly macroeconomic variables. In addition, KSIC firm codes are included by using one-hot encoding, adding 77 categorical variables. This results in a total of 167 variables (80 firm characteristics and 10 macroeconomic variables, plus 77 KSIC codes), providing a rich and detailed framework for analysis. The median of a respective month’s cross-section is used as a proxy for missing predictor values for some firms. To mitigate the impact of outliers, stock characteristics are periodically ranked in a cross-sectional analysis and then normalized to a range from -1 to 1, as described by Kelly, Pruitt, and Su (2019). This normalization ensures that extreme outliers do not disproportionately influence model prediction results. A time lag is then applied to the data to maintain the integrity of the analysis and avoid forward bias. Annual frequency data are adjusted with a 6-month lag and monthly data are adjusted with a 1-month lag to ensure the model does not use future information for prediction.

The dataset is divided into three subsets: train, validation, and test. Initially, the training set includes data from 2005 to 2011, the validation set (used for hyperparameter tuning) covers 2012 to 2014, and the testing set (used to evaluate prediction performance) includes data from a single year, 2015. Despite its time and resource-intensive nature, an expanding window method is used, for more accurate predictions. The test set moves annually to 2022, the training set expands accordingly, while the validation set remains fixed at three years. The model is retrained annually because the firm characteristics that we use as data are published annually.
3. Methodology

This section briefly describes the machine learning model and interpretable machine learning methods. The models include simple linear regression as a benchmark. Dimension reduction models such as PCR and PLS along with the Lasso-based model ENet, are utilized. Tree-based models such as RF, XGB, and LGBM are incorporated. The most complex model in our study, NN, is also used. In addition, interpretable machine learning methods such as PDP, ICE, and ALE are examined.

The excess return prediction for a stock in this research is mathematically represented as follows:

\[ r_{i,t+1} = E_t[r_{i,t+1}] + e_{i,t+1}, \]  

(1)

where \( r_{i,t+1} \) represents the return of asset \( i \) at time \( t+1 \) and \( e_{i,t+1} \) is the error term. The predictable part which we can estimate is represented as follows:

\[ r_{i,t+1}^* = E_t[r_{i,t+1}] = g(x_{i,t}) + \epsilon_{i,t+1}. \]  

(2)

where \( x_{i,t} \) is a vector of predictors at the end of month \( t \). By applying machine learning algorithms, we make no a priori assumptions about the functional form of \( g(x_{i,t}) \), although different models will indeed assume different degrees of sparsity/density and linearity/nonlinearity of the model.

Each method used in this study, from simple ordinary least squares (OLS) to more complex NNs, is trained by minimizing a given loss function. There are two types of loss functions used: mean squared error (MSE) and Huber loss. MSE is a standard choice for regression problems and focuses on minimizing the mean squared difference between the estimated values and the actual value. However, Huber loss is less sensitive to the outliers in the data than MSE. It combines the properties of both the MSE and the mean absolute error (MAE), providing a balanced approach to error minimization, particularly in the presence of anomalies in the dataset. This dual approach to loss functions allows a comprehensive evaluation of model performance under different conditions. Mathematically, the Huber loss can be defined as follows:

\[ H_\delta(a) = \begin{cases} \frac{1}{2} a^2 & \text{for } |a| \leq \delta, \\ \delta (|a| - \frac{1}{2} \delta) & \text{otherwise}, \end{cases} \]  

(3)

where \( a \) is the residual (difference between the observed and predicted values) and \( \delta \) is a threshold parameter that determines the transition point between quadratic and linear loss. This model is particularly suitable for datasets such as ours, where the presence of outliers can distort model predictions.

The first step is to estimate a linear regression model to serve as a benchmark for predicting performance. This process assumes that the relationship between the dependent variable (stock
returns) and the independent variables (predictors) can be adequately captured by a linear equation. By comparing the results of this linear model with those of more complex machine learning models, this study can assess the added value of using advanced techniques to predict stock returns. This benchmarking is critical to understanding the incremental benefits of more sophisticated models in the context of financial forecasting. The linear regression is as follows:

\[ g(x_{it}) = \beta_0 + \beta_1 x_{it} + \epsilon_{it}. \]  

(4)

ENet is a form of penalized regression that estimates the parameters of a linear model while imposing constraints to keep the size of the estimated parameters small. In practice, this involves minimizing a specific loss function that combines both L1 (LASSO) and L2 (Ridge) penalties. This approach effectively deals with multicollinearity and aids feature selection by shrinking coefficients and providing a balance between the variable selection of the lasso and the stability of the ridge in the presence of correlated predictors. The loss of ENet, which is the combined ridge and lasso loss, is as follows:

\[ L(\theta, \lambda, \alpha) = \lambda(1 - \alpha) \sum_{j=1}^{J} |\theta_j| + \frac{1}{2} \lambda \alpha \sum_{j=1}^{J} \theta_j^2, \]  

(5)

where \( J \) represents the number of predictors, \( \theta \) is the parameter vector, \( \lambda \) is the regularization parameter, and \( \alpha \) controls the balance between L1 and L2 regularization.

PCR involves first performing principal component analysis (PCA) on the predictor variables to reduce their dimensionality, and then applying linear regression to these principal components. The key mathematical expression for PCA, on which PCR is based, is as follows:

\[ X = SP^T + \epsilon, \]  

(6)

where \( X \) is the original data matrix, \( S \) the scores matrix, \( P^T \) the loadings matrix, and \( \epsilon \) the residual matrix. PLS, like PCR, reduces the predictor variables, but also considers the response variable in the reduction process. The goal is to identify the latent variables that explain both the predictors and the response. The PLS algorithm can be summarized as follows:

\[ X = S_1 P_1^T + \epsilon_1, \]  

(7)

\[ Y = S_2 P_2^T + \epsilon_2, \]  

(8)

where \( X \) and \( Y \) represent the matrices of the predictors and responses, \( S_1 \) and \( S_2 \) are the scores matrices for these predictors and responses, \( P_1 \) and \( P_2 \) are the loadings for \( X \) and \( Y \), and \( \epsilon_1 \) and \( \epsilon_2 \) are the residuals, respectively. Both PCR and PLS are effective in mitigating the curse of dimensionality and in improving model performance by focusing on the most relevant data components.

Tree-based models, which are nonparametric, differ significantly from traditional linear regression methods. Being based on decision trees, these models can handle both categorical
and continuous data and are particularly adept at capturing nonlinearity. RF is a powerful ensemble learning technique for regression and classification. It constructs multiple decision trees during training and outputs the average of these trees’ predictions for regression tasks. It also effectively handles both categorical and continuous data and is well-suited for capturing complex, non-linear relationships. In a financial context, such as predicting stock returns, RF is particularly valuable because of its ability to model complicated interactions within the data without requiring explicit specifications of the model structure. Its ensemble nature also makes it robust to overfitting, a common challenge in financial modeling. The RF model is represented mathematically as follows:

\[ g(x) = \frac{1}{B} \sum_{b=1}^{B} T_b(x|\Theta_b), \] (9)

where \( B \) is the number of trees and \( T_b(x|\Theta_b) \) is the output of the \( b \)-th tree trained on a bootstrapped sample of data, with \( \Theta_b \) representing the random selection of features used for splitting nodes in the \( b \)-th tree. Each tree \( T_b \) is built on a bootstrapped sample of the data and makes decisions based on a subset of features selected at random, enhancing model diversity and reducing variance.

XGB and LGBM share the same basic principles of gradient boosting, but there are differences in their focus and specific implementations, particularly with respect to tree construction, regularization, and optimization techniques. Both models aim to minimize a similar type of objective function, which combines a loss function that measures prediction error and regularization term to control model complexity. The XGB model and LGBM model are as follows:

\[ g(x) = \sum_{b=1}^{B} \eta \cdot f_b(x) + \Omega(f_b), \] (10)

where \( f_b(x) \) is the output of the \( b \)-th tree, \( \eta \) the learning rate, and \( \Omega(f_b) \) the regularization term applied to the \( b \)-th tree. To further detail the loss function of the two models as follows:

\[ \min_{f_b}(\sum_{i=1}^{N} L(y_i, g(x_i)) + \sum_{b=1}^{B} \Omega(f_b)), \] (11)

where \( L(y_i, g(x_i)) \) is the loss function that measures the difference between the actual value of \( y_i \) and the output \( g(x_i) \) and \( \Omega(f_b) \) is the regularization term for the \( b \)-th tree. There are differences between XGB and LGBM. XGB typically uses a depth-wise approach, ensuring balanced tree growth. On LGBM uses a leaf-wise tree growth strategy, which often results in deeper, more complex trees but aims to reduce loss more rapidly. Both models include regularization in their loss function, and the specifics of how regularization is applied and formulated differ. XGB has a detailed focus on regularization, including both the structure of the tree and the value of the leaf weights in its regularization term. LGBM also considers these aspects but is optimized for speed and efficiency, especially in how it handles data and computes gradients. There are no fundamental differences in the mathematical expression of the objective functions between XGB and LGBM, the difference lies in the details of
optimization and the specific strategies used for tree construction and handling data. These differences can lead to variations in model performance, training speed, and computing usage.

The final and most complex model used in this study is a basic form of NN: the feed-forward network. This model comprises an input layer that receives input values, one or more hidden layers, and an output layer that delivers the result. We consider models with up to five hidden layers:

\[ Z_j = \beta_0 + \sum_{i=1}^P X_i. \]  

(12)

A nonlinear activation function, \( Z_j^* \), is applied in our study:

\[ Z_j^* = ReLU(Z_j), \]  

(13)

where \( ReLU \) is defined as \( ReLU = max(0, Z_j) \).

The predicted result is then a linear combination of the activated nodes:

\[ Y = \alpha_0 + \sum_{j=1}^N \alpha_j Z_j^*, \]  

(14)

where there are \( p \) predictors and 1 hidden layer with \( N \) nodes. Therefore, our NN works based on this formula. To fine-tune this complex model, we implement four regularization techniques: learning rate adjustment, early stopping, batch normalization, and ensemble methods. These techniques help optimize network performance and mitigate overfitting, thus ensuring robust predictions.

The hyperparameter combinations for all models are identified using grid-search methods. Table 2 presents the details of the hyperparameter used in this study.

[Table 2 inserted about here]

In this study, we use the out-of-sample \( R^2 \) metric to evaluate the predictive performance of different machine learning models across firms and time periods. This approach provides a panel-level evaluation of each model. The out-of-sample \( R^2 \) is calculated as follows:

\[ R^2_{oos} = 1 - \frac{\sum_{(i,t) \in (\tau)} (r_{i,t+1} - r_{i,t+1})^2}{\sum_{(i,t) \in \tau} r_{i,t+1}^2}, \]  

(15)

where \( \tau \) represents the test set, which is different from the data used for train or validation in

\[ \text{footnote} 4 \]

For the use of other models in Korean market research, refer to the recent studies of Bang and Ryu (2023), Kim, Cho, and Ryu (2021a, 2021b, 2022, 2023), Kim and Enke (2018), and Park and Ryu (2021).
hyperparameter settings. $f_{t,t+1}$ is the predicted monthly return. This method ensures that the evaluation focuses on the accuracy of models in predicting return in an unseen dataset, providing a robust measure of its predictive ability.

Despite their robust out-of-sample predictive performance, machine learning models often face the challenge of being black-box models, which makes interpretation difficult. This issue is particularly pertinent in financial datasets, which have more complex and intricate relationships than the datasets used in traditional machine learning tasks, such as image or text analysis. To address this issue, we explore the relationship between predictors and predicted values using three methods. Starting with the relatively simple PFI, we move to more advanced interpretability tools, such as PDP, ICE, and ALE. These visualization techniques enhance the understanding of machine learning models and provide the basis for an empirical analysis that provides interpretability.

The first interpretative method that we explore for understanding variable importance in machine learning is PFI. This approach aims to identify variables that influence return prediction while controlling for other predictors. However, it is a simple method that does not require retraining the model. Instead, it involves randomly shuffling the values of a variable and then comparing the predicted outcome to the original prediction. If the out-of-sample $R^2$ decreases with the shuffled data, the variable has a significant impact on the prediction. While this method does not provide detailed insights into the process, it effectively identifies variables that have the most significant impact on predictions. We thus use PFI to identify the most important predictive variables and interpret how they affect stock returns using interpretable machine learning techniques.

The PDP shows the marginal effect of one or two variables on the predicted value estimated by the machine learning model. It visualizes the relationship between the predicted value and a variable. For a variable $X_s$, the PDP is expressed as follows:

$$PD_s(x_s) = \frac{1}{N} \sum_{i=1}^{N} f(x_s, X_{i,-s}),$$

where $x_s$ is the specific value of the variable $X_s$; $f(x_s, X_{i,-s})$ denotes the prediction of the model for $x_s$ and other features $X_{i,-s}$ in the dataset, excluding $X_s$ and $N$ is the total number of instances. PDP is intuitive and provides a clear interpretation. However, it requires the assumption of independence, which is rare in complex contexts such as financial markets. In recognition of this limitation, the study employs additional visualization tools.

The ICE plot extends the idea of PDP by showing how predictions change for individual observations. Unlike the PDP, which provides average effects, ICE plots provide a detailed visualization. The ICE for $X_s$ is defined as follows:

$$ICE_{i,s}(x_s) = f(x_s, X_{i,-s}),$$

where $x_s$ is the specific value of $X_s$ and $f(x_s, X_{i,-s})$ denotes the prediction of the model for $x_s$ and the set of other features $X_{i,-s}$ for the $i$th instance, excluding $X_s$. ICE plots show the model’s response to variations in a variable for different instances, which can be difficult to interpret.
with PDP alone. By analyzing both PDP and ICE, one can understand the average trends and individual variations in predictions. Zhao and Hastie (2021) show that using PDP and ICE can extract causal information from black-box models under three strong conditions: proximity to a natural function, satisfying the backdoor criterion and using visualization tools. The first condition is addressed by the ability of machine learning to approximate natural functions, while the second acknowledges the impact of firm characteristics and macroeconomic variables on stock returns. However, in the real world, especially in financial markets, strong independence assumptions are difficult to satisfy. Despite these limitations, the visualization through PDP and ICE provides insights into the interpretation of black-box models, providing valuable insights into financial market analysis with machine learning.

The ALE method explains how a feature generally affects the predictions of a machine learning model. ALE mitigates the problem of bias found in PDP. ALE is derived using the following formula:

\[ ALE_s(x_s) = \int_{x_{s,\text{min}}}^{x_s} (E[f(X)|X_s = z] - E[f(X)])dz, \]

where \(x_{s,\text{min}}\) and \(x_s\) are the minimum and specific values of \(X_s\), \(E[f(X)|X_s = z]\) is the expected value of the model prediction when given the value \(z\) of \(X_s\), and \(E[f(X)]\) is the overall expected value of the model prediction. ALE, as a global method, provides insights into the overall effects of a predictor across an entire range of data. Specifically, it calculates the cumulative differences between the local conditional expectation and the overall expectation, avoiding biases in PDP due to feature correlations. ALE is particularly valuable for understanding the global structure of the data and how the variations in a predictor affect a model’s average predictions.

4. Empirical results

In our empirical analysis, out-of-sample \(R^2\) value is used to illustrate the comparison of different machine-learning models. The results with variable machine learning models are shown in Table 3. The simple OLS model with Huber loss shows poor out-of-sample performance, dominated by the simplistic assumption of zero stock returns for all firms in all periods. ENet, a model with an additional penalty term, demonstrates improvement. Dimensional reduction methods, especially PCR, show significant improvements in out-of-sample \(R^2\). Non-linear methods, such as tree-based and neural network models, generally show better predictive performance. Among them, tree-based models (i.e., RF, XGB, LGBM) perform exceptionally well in the Korean market. Despite its complexity, the neural network model does not outperform the tree-based models, possibly indicating the need for more extensive data.

Table 3 shows that the overall machine-learning model we used has a lower out-of-sample \(R^2\) compared to the results reported for developed markets. This suggests that machine learning
models may not be effective in emerging markets where the data period is shorter. The tree-based models outperformed the neural network model which requires more complexity. It is challenging to apply research from developed markets, where more complex models are studied, to emerging markets (Chen, Pelger, and Zhu, 2024).

In Figure 1, we investigate the optimal complexity hyperparameters in view of the complexity of various machine learning models. Interpreting complexity over short sample periods can be challenging. Rapid and inconsistent changes in complexity suggest that the signal-to-noise ratio is difficult to detect in the Korean market. Overall, nonlinear models with superior out-of-sample $R^2$ tended to select more complex hyperparameters. The results for NNs in the Korean market differ from the observations in developed markets, where shallow models outperformed complex ones. This suggests a market-specific variance in model performance.

[Figure 1 inserted about here]

Next, we investigate how different variables behave in the machine learning predictions. First, we use the PFI method to identify the key variables that contribute to machine learning stock return prediction in the Korean market. This approach will help us understand the dynamics of the Korean financial market and the variables that are most influential in machine learning stock return prediction, as shown in Figures 2 and 3. The results of this analysis are crucial to understanding which variables are important in predicting stock returns.

[Figure 2 inserted about here]

[Figure 3 inserted about here]

The PFI analysis shows that, in the Korean market, the 36-month momentum ($mom_{36m}$) is the most significant variable in all models except the NN. Its importance is evident even in models such as XGB and LGBM, which are less affected by variable shuffling, as shown in Figures 2 and 3. The top 10 variables based on PFI across all models, as shown in Figure 3, reveal a prevalence of momentum-related variables, including 36-month momentum ($mom_{36m}$), momentum change ($chmom$), 1-month momentum ($mom_{1m}$), and industry-adjusted momentum ($indmom$). Notably, $sic2$, the industrial classification code, is more important in the Korean market than in other markets, possibly due to significant industry-specific differences in this market.

Further analysis of the strong correlation between stock returns and the 36-month momentum is conducted using visualization methods other than PFI. Following the approach of Zhao and Hastie (2021), PDP and ICE are used to derive meaningful interpretations from machine learning models. While machine learning stock return forecasting is not flawless, it is designed to approximate natural functions. However, it has limitations, especially considering that momentum is not completely independent of other variables. Recognizing these limitations is critical to fully understand the model’s predictive capabilities. The PDP and ICE plots are demonstrated by the nonlinear models shown in Figure 4.
In Figure 4, the tree-based models show similar patterns across all instances, while the NN model shows significantly different values for each instance. This means that the interpreter NN is not well trained. For models other than NN, the 36-month momentum had a dismal effect on the overall observations beyond a certain point (-0.50 in RF, -0.25 in XGB, and -0.25 in LGBM). This suggests that the 36-month momentum is an important factor in predicting stock returns with machine-learning approaches. Furthermore, the ICE plots for tree models followed a similar course to the PDP plots, indicating that PDP effectively captures the relationship between stock returns and the 36-month momentum.

Next, we show the ALE plots in Figure 5 for further analysis, addressing a key limitation of the PDP, that is, the potential for misleading results in the presence of strong correlations between predictors. ALE plots provide a more accurate interpretation of these scenarios by focusing on the local effects of predictors, thus avoiding the biases introduced by the global average effects considered in PDP.

The ALE plot in Figure 5 confirms the results of the PDP and ICE plots in Figure 4. As observed in tree models, the impact of the 36-month momentum on stock returns diminishes after a certain point. This trend is also present, albeit less pronounced, in the NN model, which shows a decline in the impact of the 36-month momentum after a certain threshold. This finding helps us understand how the most important variable, the 36-month momentum, performs in machine learning prediction.

In this section, we perform out-of-sample predictions of stock returns using several machine-learning methods. Our results are generally low in out-of-sample $R^2$ compared to those reported for developed markets, and the choice of hyperparameter is inconsistent. Also, models that are too complex (NN) are not well-trained. This suggests that the datasets from emerging markets are not large enough to build good predictive models compared to datasets from studies conducted in developed markets. According to Kelly, Malamud, and Zhu (2024), more complex models can produce higher return predictions, but this implies that data availability is required for more complex models. Also, we examine the behavior of machine learning models with respect to the most important variable identified by PFI, the 36-month momentum. Using methods such as PDP, ICE, and ALE, we find that, after a certain point, the 36-month momentum does not contribute significantly to stock returns. This approach to interpreting machine learning models provides insights beyond the mere impact of the 36-month momentum on predictions. It also allows us to observe how changes in the 36-month momentum affect stock return predictions of machine learning. Such interpretations enhance our understanding of machine learning processes by revealing how specific firm characteristics or macroeconomic variables affect stock returns.

5. Conclusions
This study constructs a dataset of firm characteristics and macroeconomic variables and then compares various machine learning models to determine the most effective one for predicting stock returns in the Korean market. Unlike in developed markets, tree-based models can be more effective than deep learning models in emerging markets with smaller datasets. This means that emerging markets yield different results from developed market research which use more complex models. We also analyze the most important variables in the Korean market using the PFI method and find that, as in many markets, momentum-related variables are significant. In particular, the 36-month momentum is the most significant variable in almost all the models. We further analyze the key variable, 36-month momentum using several machine learning interpretation methods, including PDP, ICD, and ALE. This study thus shows that interpretable methods can help us better understand stock return prediction via machine learning models.

References


**Figure 1** Time-varying model complexity

Panel A. PLS complexity

Panel B. PCR complexity

Panel C. ENet complexity

Panel D. RF complexity

Panel E. XGB complexity

Panel F. LGBM complexity

Panel G. NN complexity

**Notes:** This figure reports the complexities of machine learning models from 2015 to 2022. Complexity is reported in terms of the number of components for dimensional reduction models PCR and PLS in Panel A and B, the alpha parameter for regularized linear model ENet in Panel C, the maximum depth for RF in Panel D, a bagging model among tree-based models, and the number of estimators for boosting series for XGB and LGBM in Panel E and F. It also reports the optimal number of layers for NN models in Panel G.
**Figure 2** Key variables calculated by permutation methods

Panel A. PLS PFI

Panel B. PCR PFI

Panel C. ENet PFI
Notes: This figure shows the PFI for the top 20 most influential variables in the machine learning models in panels A to G. The PFI value is an average of the entire training sample.
Figure 3 Key variables sorted by PFI of all models.

Notes: This figure shows the PFI importance for the top 20 most influential variables across all machine learning modes, with the most influential attributes in the overall model located at the top. The color gradient within each column indicates the most influential (blue) to the least influential (white) variables.
**Figure 4** Visualization of relationships between key 36-month momentum and stock returns with PDP and ICE

Panel A. RF

Panel B. XGB

Panel C. LGBM

Panel D. NN

Notes: PDP plots and ICE plots for 36-month momentum normalized to the range [-1, 1] and stock returns in the RF, XGB, LGBM, and NN models in panels A to D.
**Figure 5** Visualization of relationships between 36-month momentum and stock returns with ALE

Panel A. RF ALE

Panel B. XGB ALE

Panel C. LGBM ALE

Panel D. NN ALE

Notes: ALE plots for 36-month momentum normalized to the range [-1, 1] and stock returns in the RF, XGB, LGBM, and NN models in panels A to D.
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
<th>Frequency</th>
<th>Studies</th>
<th>Characteristic definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>abacc</td>
<td>Absolute accruals</td>
<td>Annual</td>
<td>Bandyopadhyay et al. (2010)</td>
<td>Absolute value of acc</td>
</tr>
<tr>
<td>acc</td>
<td>Working capital accruals</td>
<td>Annual</td>
<td>Sloan (1996)</td>
<td>((ib-oancf)/((at+\text{lag}(at))/2))</td>
</tr>
<tr>
<td>age</td>
<td>Firm age</td>
<td>Annual</td>
<td>Jiang et al. (2005)</td>
<td>Firm age is defined as the number of months between event month t and the first month that a stock appears in FnGuide</td>
</tr>
<tr>
<td>agr</td>
<td>Asset growth</td>
<td>Annual</td>
<td>Cooper et al. (2008)</td>
<td>((at/\text{lag}(at))-1)</td>
</tr>
<tr>
<td>beta</td>
<td>Beta</td>
<td>Annual</td>
<td>Fama and MacBeth (1973)</td>
<td>Beta provided by FnGuide</td>
</tr>
<tr>
<td>bm_x</td>
<td>Book to market</td>
<td>Annual</td>
<td>Rosenberg et al. (1985)</td>
<td>(\text{ceq/mve}_f)</td>
</tr>
<tr>
<td>bm_ia</td>
<td>Industry adjusted book-to-market</td>
<td>Annual</td>
<td>Asness et al. (2000)</td>
<td>(\text{indadj}(bm))</td>
</tr>
<tr>
<td>capxint</td>
<td>Capital expenditures to asset</td>
<td>Annual</td>
<td>-</td>
<td>(\text{capx}/((at+\text{lag}(at))/2))</td>
</tr>
<tr>
<td>cash</td>
<td>Cash Holdings</td>
<td>Annual</td>
<td>Palazzo (2012)</td>
<td>(\text{che/at})</td>
</tr>
<tr>
<td>cashdebt</td>
<td>Cash flow to debt</td>
<td>Annual</td>
<td>Ou and Penma (1989)</td>
<td>((ib+dpa)/((lt+\text{lag}(lt))/2))</td>
</tr>
<tr>
<td>cashpr</td>
<td>Cash productivity</td>
<td>Annual</td>
<td>Chandrashekar and Rao (2009)</td>
<td>((\text{mve}_f+\text{ltd–at})/\text{che})</td>
</tr>
<tr>
<td>cf</td>
<td>Cash flow</td>
<td>Annual</td>
<td>-</td>
<td>((ib-((\text{act}\text{-lag}(\text{act}))-(\text{che}\text{-lag}(\text{che})))-(\text{lct}\text{-lag}(\text{lct}))-(\text{dlc}\text{-lag}(\text{dlc}))-(\text{txp}\text{-lag}(\text{txp}))-(\text{dpa})))/((at+\text{lag}(at))/2))</td>
</tr>
<tr>
<td>cfp</td>
<td>Cash flow to price ratio</td>
<td>Annual</td>
<td>Desai et al. (2016)</td>
<td>(\text{oancf/mve}_f)</td>
</tr>
<tr>
<td>cfp_ia</td>
<td>Industry adjusted cash flow to price ratio</td>
<td>Annual</td>
<td>Asness et al. (2000)</td>
<td>(\text{indadj}(cfp))</td>
</tr>
<tr>
<td>cfroa</td>
<td>Cash flow return on assets</td>
<td>Annual</td>
<td>-</td>
<td>(\text{oancf}/((at+\text{lag}(at))/2))</td>
</tr>
<tr>
<td>chadv</td>
<td>Change of advertisement</td>
<td>Annual</td>
<td>-</td>
<td>(\text{log}(1+xad)-\text{log}((1+\text{lag}(xad))))</td>
</tr>
<tr>
<td>chato</td>
<td>Asset turnover</td>
<td>Annual</td>
<td>-</td>
<td>((\text{sale}/((at+\text{lag}(at))/2))-(\text{lag(sale)}/((\text{lag(at)}+\text{lag2(at)}/2))))</td>
</tr>
</tbody>
</table>

26
<table>
<thead>
<tr>
<th>RAW TEXT</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>chatoia</strong></td>
<td>Industry-adjusted change in asset turnover</td>
</tr>
<tr>
<td><strong>chcsho</strong></td>
<td>Change in shares outstanding</td>
</tr>
<tr>
<td><strong>chempia</strong></td>
<td>Industry adjusted change in employees</td>
</tr>
<tr>
<td><strong>chinv</strong></td>
<td>Change in inventory</td>
</tr>
<tr>
<td><strong>chmom</strong></td>
<td>Change in 6-month momentum</td>
</tr>
<tr>
<td><strong>chpm</strong></td>
<td>Profit margin</td>
</tr>
<tr>
<td><strong>chomia</strong></td>
<td>Industry adjusted change in profit margin</td>
</tr>
<tr>
<td><strong>chtex</strong></td>
<td>Change in tax expense</td>
</tr>
<tr>
<td><strong>currat</strong></td>
<td>Current ratio</td>
</tr>
<tr>
<td><strong>depr</strong></td>
<td>Depreciation/PP&amp;E</td>
</tr>
<tr>
<td><strong>divi</strong></td>
<td>Dividend initiation</td>
</tr>
<tr>
<td><strong>divo</strong></td>
<td>Dividend omission</td>
</tr>
<tr>
<td><strong>dolvol</strong></td>
<td>Won trading volume</td>
</tr>
<tr>
<td><strong>dy</strong></td>
<td>Dividend to price</td>
</tr>
<tr>
<td><strong>egr</strong></td>
<td>Growth in common shareholder equity</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>ep_x</td>
<td>Earning to price</td>
</tr>
<tr>
<td>gma</td>
<td>Gross profitability</td>
</tr>
<tr>
<td>grgw</td>
<td>Growth of goodwill</td>
</tr>
<tr>
<td>grcapx</td>
<td>Growth of capital expenditures</td>
</tr>
<tr>
<td>herf</td>
<td>Industry sale concentration</td>
</tr>
<tr>
<td>hire</td>
<td>Employ grow rate</td>
</tr>
<tr>
<td>indmom</td>
<td>Industry momentum</td>
</tr>
<tr>
<td>indsale×</td>
<td>Industry adjusted sale</td>
</tr>
<tr>
<td>invest</td>
<td>Capital expenditures and inventory</td>
</tr>
<tr>
<td>lgr</td>
<td>Growth in long-term debt</td>
</tr>
<tr>
<td>mom1m</td>
<td>1-month momentum</td>
</tr>
<tr>
<td>mom12m</td>
<td>12-month momentum</td>
</tr>
<tr>
<td>mom36m</td>
<td>36-month momentum</td>
</tr>
<tr>
<td>mom6m</td>
<td>6-month momentum</td>
</tr>
<tr>
<td>mve_ia</td>
<td>Industry adjusted size</td>
</tr>
<tr>
<td>mvel1</td>
<td>Size</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>operprof</td>
<td>Operating profitability</td>
</tr>
<tr>
<td>pchcapx</td>
<td>Percent change in capital expenditure</td>
</tr>
<tr>
<td>pchcapx_ia</td>
<td>Industry adjusted change in capital expenditures</td>
</tr>
<tr>
<td>pchcurrat</td>
<td>Percent change in current ratio</td>
</tr>
<tr>
<td>pchdepr</td>
<td>Percent change in depreciation</td>
</tr>
<tr>
<td>pchgm_pchsale</td>
<td>Percent change in gross margin – sale</td>
</tr>
<tr>
<td>pchquick</td>
<td>Percent change in quick</td>
</tr>
<tr>
<td>pchsale_pchinv</td>
<td>Percent change in sale – inventory</td>
</tr>
<tr>
<td>pchsale_pchrect</td>
<td>Percent change in sale receivables</td>
</tr>
<tr>
<td>pchsale_pchxs</td>
<td>Percent change in sale – xsga</td>
</tr>
<tr>
<td>pchsaleinv</td>
<td>Percent change sales to inventory</td>
</tr>
<tr>
<td>pctacc</td>
<td>Percent accruals</td>
</tr>
<tr>
<td>quick</td>
<td>Quick ratio</td>
</tr>
<tr>
<td>rd</td>
<td>R&amp;D to sale</td>
</tr>
<tr>
<td>rdbias</td>
<td>Bias of R&amp;D</td>
</tr>
<tr>
<td>rdsale</td>
<td>R&amp;D to sale</td>
</tr>
<tr>
<td>Characteristic</td>
<td>Definition</td>
</tr>
<tr>
<td>---------------</td>
<td>------------</td>
</tr>
<tr>
<td>roa</td>
<td>Return on asset</td>
</tr>
<tr>
<td>roe</td>
<td>Return on equity</td>
</tr>
<tr>
<td>roic</td>
<td>Return on invested capital</td>
</tr>
<tr>
<td>rsup</td>
<td>Revenue surprise</td>
</tr>
<tr>
<td>sacc</td>
<td>Accrual to sale</td>
</tr>
<tr>
<td>salescash</td>
<td>Sale to cash</td>
</tr>
<tr>
<td>salesinv</td>
<td>Sales to inventory</td>
</tr>
<tr>
<td>salerec</td>
<td>Sales to receivables</td>
</tr>
<tr>
<td>secured</td>
<td>Secured debt</td>
</tr>
<tr>
<td>securedind</td>
<td>Secured debt indicator</td>
</tr>
<tr>
<td>sgr</td>
<td>Sales growth</td>
</tr>
<tr>
<td>sp</td>
<td>Sale to price</td>
</tr>
<tr>
<td>tang</td>
<td>Debt capacity to firm tangibility</td>
</tr>
<tr>
<td>turn</td>
<td>Share turnover</td>
</tr>
<tr>
<td>xadint</td>
<td>Advertisement to assets</td>
</tr>
<tr>
<td>xrdint</td>
<td>R&amp;D to assets</td>
</tr>
</tbody>
</table>

**Notes:** This table shows the available firm characteristics converted to Korean Market data listed by Green, Hand, and Zhang (2017) and additional firm characteristics. The acronyms in the column definition of characteristics are defined as: act: total current asset; ap: account payable; at: total asset; capx: capital expenditures; ceq: total common equity; che: cash and short-term investments; cogs: cost of goods sold; csho: common shares outstanding; cshrc: common shares reserved for conversion—converted debt; dcvt: convertible debt; dltt: total long-term debt; dm: mortgages and other secured debt; dpa: depreciation and amortization; dt: total dividends; ebit: earnings before interest and taxes;
ebitda: earnings before interest; emp: number of employees; gdwl: goodwill; ib: income before extraordinary items; intan: total intangible assets; invt: total inventories; lct: total current liabilities; lt: total liabilities; mve_f: firm size; ni: net income; nopi: nonoperating income; oanef: operating activities—net cash flow; ppent: total property, plant, and equipment; revt: revenue; sale: sale; shtout: share outstanding spi: special items; txt: total income tax; vol: volume; xad: advertising expense; xint: total interest and related expense; xrd: research and development expense; xsga: selling, general, and administrative expense. The operator “indaj” means industrial adjustment.
<table>
<thead>
<tr>
<th>Model</th>
<th>Huber loss</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>OLS+H</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>PLS</td>
<td>-</td>
<td>K∈(1, 50)</td>
</tr>
<tr>
<td>PCR</td>
<td>O</td>
<td>K∈(1, 50)</td>
</tr>
<tr>
<td>ENet</td>
<td>O</td>
<td>λ∈(10^{-4}, 10^{-1})</td>
</tr>
<tr>
<td>RF</td>
<td>-</td>
<td>Depth∈(1, 8)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trees=[10, 50, 100, 250, …]</td>
</tr>
<tr>
<td>XGB</td>
<td>-</td>
<td>Depth∈{1, 2, 3}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trees=[10, 50, 100, 250, …]</td>
</tr>
<tr>
<td>LGBM</td>
<td>-</td>
<td>Depth∈{1, 2, 3}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trees=[10, 50, 100, 250, …]</td>
</tr>
<tr>
<td>NN</td>
<td>O</td>
<td>Layer∈{1, 2, 3, 4, 5}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Batch size=1,000</td>
</tr>
</tbody>
</table>

Notes: The table describes the hyperparameters tuned in each machine-learning model with a grid search. The models considered include Ordinary Least Square with Huber loss (OLS+H), Partial Least Suares Regression (PLS), Principal Composition Regression, ElasticNet (Enet), Random Forest (RF), eXtreme Gradient Boosting (XGB), Light Gradient Boosting Machine (LGBM), and neural network (NN) with layers 1 to 5.
Table 3 Out-of-sample prediction performance

<table>
<thead>
<tr>
<th></th>
<th>OLS+H</th>
<th>PLS</th>
<th>PCR</th>
<th>ENet</th>
<th>RF</th>
<th>XGB</th>
<th>LGBM</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>-18.73</td>
<td>-2.16</td>
<td>-0.07</td>
<td>-5.11</td>
<td>0.31</td>
<td>0.08</td>
<td>0.18</td>
<td>0.14</td>
</tr>
<tr>
<td>Top70</td>
<td>-9.08</td>
<td>-2.11</td>
<td>0.00</td>
<td>-3.23</td>
<td>0.29</td>
<td>0.22</td>
<td>-0.07</td>
<td>-0.02</td>
</tr>
<tr>
<td>Bot30</td>
<td>-1.21</td>
<td>-0.47</td>
<td>-0.01</td>
<td>-0.65</td>
<td>-0.14</td>
<td>-0.71</td>
<td>-0.50</td>
<td>-0.05</td>
</tr>
</tbody>
</table>

Notes: This table reports the out-of-sample prediction of monthly returns from 2015 to 2022 as the out-of-sample $R^2$ percentage of monthly returns in the full sample, the sample including firms with the top 70% of market values, and the sample including the firms with the bottom 30% of market values. The models considered include Ordinary Least Square with Huber loss (OLS+H), Partial Least Squares Regression (PLS), Principal Composition Regression, ElasticNet (Enet), Random Forest (RF), eXtreme Gradient Boosting (XGB), Light Gradient Boosting Machine (LGBM), and neural network (NN) with layers 1 to 5.