Routledge
Taylor & Francis Group

Quantitative Finance

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/rquf20

Forecasting the equity premium: can machine learning beat the historical average?

Xingfu Xu & Wei-han Liu

To cite this article: Xingfu Xu & Wei-han Liu (2024) Forecasting the equity premium: can machine learning beat the historical average?, Quantitative Finance, 24:10, 1445-1461, DOI: 10.1080/14697688.2024.2409278

To link to this article: https://doi.org/10.1080/14697688.2024.2409278

Quantitative Finance, 2024
Vol. 24, No. 10, 1445–1461, https://doi.org/10.1080/14697688.2024.2409278

Forecasting the equity premium: can machine learning beat the historical average?

XINGFU XU†‡ and WEI-HAN LIU †§*

†Department of Finance, Southern University of Science and Technology, yuan Avenue, Shenzhen, 518055, People's Republic of China
‡School of Accounting and Finance, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, People's Republic of China
§School of Digital Economy and Management, Fuyao University of Science and Technology, Nanyu Town, Fuzhou High tech Zone, Fuzhou, Fujian, People's Republic of China

(*Received 22 March 2023; accepted 21 August 2024*)

We empirically predict the equity premium with the selected machine learning methods in Gu *et al*. (Empirical asset pricing via machine learning. *Rev. Financ. Stud.*, 2020, **33**(5), 2223–2273). We also consider four additional popular machine learning methods (ridge regression, support vector regression, k-nearest neighbors, and extreme gradient boosted trees) and their combination method. Using a dataset of both macroeconomic and technical predictors, we find that despite showcasing strong in-sample forecasting abilities, particularly with tree-based models, the out-of-sample results support Welch and Goyal (A comprehensive look at the empirical performance of equity premium prediction. *Rev. Financ. Stud.*, 2008, **21**(4), 1455–1508) that the competing forecasting models generally fail to outperform the historical average benchmark. We attribute this failure to the small dataset size and the low signal-to-noise ratio inherent in equity premium prediction. Our variable importance analysis further identifies three bond interest rate-related variables as the most dominant predictors for the equity premium. The economic value from a market timing perspective highlights that the historical average benchmark strategy generates the highest average return of 25.63% and the best Sharpe ratio of 0.8. Finally, our findings are robust across a variety of settings.

Keywords: Machine learning; Equity premium; Historical average

1. Introduction

Getting more accurate predictions is one of the major challenges for academic research and industry practice. Stock returns or the equity risk premium is noticed for their volatility and uncertainty, and predicting stock returns or the equity risk premium is of great interest and great challenges to both academia and the finance industry. It has been a long-standing debate about whether we can develop empirical models to provide a more accurate forecast of the U.S. equity premium than the historical average (HA) benchmark (Spiegel 2008). The accurate prediction of equity premiums has important implications for the interpretation of asset pricing models and the development of investment strategies. Finance academics also require reliable stock return predictions to evaluate market efficiency. That is, a comprehensive understanding of stock returns can help develop more realistic asset pricing models to explain the real markets (Rapach and Zhou 2013). Meanwhile, portfolio managers need accurate stock return forecasts to better allocate their assets. It is imperative to develop various complex models to generate more accurate stock returns with an implicit assumption that model complication breeds improvement in prediction performance.

While historical mean functions like a straightforward and powerful benchmark, a strand of studies resort to various sophisticated strategies such as restrictions of steady-state valuation (Campbell and Thompson 2008), forecast combination (Rapach *et al*. 2010), regime shifts (Henkel *et al*. 2011), technical indicators (Neely *et al*. 2014), and ridgeless regression (Kelly *et al*. 2024). However, Dichtl *et al*. (2021) conclude that most complicated attempts fail to outperform the HA benchmark out-of-sample after accounting for data snooping. Other prior studies present mixed conclusions (Welch and Goyal 2008, Ferreira and Santa-Clara 2011, Dangl and Halling 2012, Timmermann 2018). In short, we still cannot reach a definite conclusion since Welch and Goyal (2008)

*Corresponding author. Email: weihanliu2002@yahoo.com compared the performance between historical mean and other complicated methods in terms of predicting the equity premium. Then several empirical studies join the pursuit and develop even more complex models to improve forecasting performance in terms of statistically and economically significant out-of-sample gains relative to the historical mean benchmark. All these efforts remind us of several fundamental issues. For example, does it pay off to try complicated techniques? What are the key predictors in determining the equity premium?

Some literature typically assumes a linear relationship between equity premium and its predictors. However, nonlinear and time-varying relationships are present at least due to structural breaks and model instability (Rapach *et al.* 2010, Pettenuzzo and Timmermann 2011). Our study plans to investigate those interesting issues by employing the lately widely used machine learning methods to horserace predictive ability. Machine learning is empirically confirmed for its strong predictive power in forecasting cross-sectional returns and capturing the complex relationships between the target variable and predictors without imposing prior model assumptions (Gu *et al.* 2020, Akbari *et al.* 2021, Leippold *et al.* 2022). While a very limited number of studies employ machine learning methods to time-series return forecasts at the aggregate market level (Bianchi *et al.* 2021), we underline that machine learning methods have noteworthy advantages over conventional linear regression models.

In terms of dealing with a large number of potential predictors with an emphasis on dimension reduction techniques, machine learning methods can be a direct solution (Gu *et al.* 2020). This popular adoption has gained momentum, and some authors conclude that a machine learning method is a promising tool, especially for extraordinarily complex and challenging tasks. However, some questioning voices are not well considered. Meanwhile, some other previous studies alternatively concluded the historical mean is a straightforward but stringent benchmark (Goyal and Welch 2003, Welch and Goyal 2008, Goyal *et al.* 2024). As these methods compete over forecasting the equity premium, we attempt to join the discussion with more refined detailed analyses. We expect our horserace can also shed light on whether and how machine learning methods can beat the HA forecast and find out the significant predictors.

We follow Gu *et al.* (2020) and start with the simple linear regression and the employed machine learning methods to test their predictive power. Then we extend the selection of machine learning methods with four other popular machine learning algorithms. We select these four more machine learning methods because they are not used to predict stock returns but demonstrate their strong predictive power in other financial predictions in existing studies (Holopainen and Sarlin 2017, Akbari e al. 2021). These four other competing methods are ridge regression (Ridge), support vector regression (SVR), k-nearest neighbors (KNN), and extreme gradient boosted trees (XGBoost). We also follow Timmermann (2006) and Rapach *et al.* (2010) and consider their combination method that averages predictions of these individual predictive models.

We believe that the dataset itself and the included variables also play their respective important roles. For the sake of brevity, we consider the predictive ability of two major groups of variables: macroeconomic variables (Welch and Goyal 2008) and technical indicators (Neely *et al.* 2014). While macroeconomic variables depict the business cycles to reflect the market fundamentals, Neely *et al.* (2014) document that technical indicators can deliver additional information and improve forecasting ability. We employ the in-sample (out-of-sample) R-squared and success ratio as our two evaluation criteria. Thus, we conduct detailed in-sample and out-of-sample analyses to compare the performance of the employed machine learning methods with the HA benchmark.

Furthermore, we also aim to improve the horserace via the following attempts. We separately evaluate the out-of-sample predictive ability of macroeconomic versus technical indicators. To account for the heavy-tailed distribution of financial returns, we employ a median absolute error scoring function on the validation set. To identify the key predictors in forecasting the equity premium, we utilize the variable importance technique (Gu *et al.* 2020), unveiling the significant variables within each group. In addition, we explore the practical application of these machine learning forecasts from a market timing perspective by comparing their economic value against two benchmark strategies: the HA strategy and the buy-and-hold strategy.

In brief, our attempts make three contributions based on refining the study of Wolff and Neugebauer (2019). First, their paper focuses exclusively on tree-based machine learning models. Ours adopts a wider variety of machine learning methods such as tree-based models, neural networks, SVR, and KNN. Second, Neely *et al.* (2014) document that technical indicators deliver statistically and economically significant predictive ability both in-sample and out-of-sample. We consider the role of technical indicators in forecasting the equity premium, but Wolff and Neugebauer (2019) did not consider technical indicators. Finally, our entire sample covers a long period from 1926:12–2020:12, comparable to the previous studies such as Welch and Goyal (2008).

Our study concluded that although most machine learning methods, especially tree-based models, exhibit strong predictive power in in-sample forecasting, they fail to significantly beat the HA benchmark in out-of-sample prediction. Our results are in line with the conclusions of Welch and Goyal (2008) that those equity premium forecasts based on the employed sophisticated models fail to beat the HA benchmark. The market timing performance also demonstrates that the HA benchmark strategy can deliver the largest economic value in terms of average annual return and Sharpe ratio. We attribute the underperformance of machine learning in equity premium prediction to the insufficiently large datasets and the low sign-to-noise ratio. Our variable importance analysis identifies three bond interest rate-related variables as the most important predictors for macroeconomic variables, whereas the results for technical indicators are inconclusive. Furthermore, our conclusions are sustained in various robustness checks such as different rolling window estimation schemes, alternative forecasting window sizes, forecasting simple equity premiums, alternative validation schemes for hyperparameter tuning, adding newly identified variables, and alternative macroeconomic datasets.

While most previous literature holds a positive evaluation or even overpraises machine learning methods, our study is not the first piece to be critical of machine learning methods. For example, the simple method does not necessarily underperform and Beutel *et al.* (2019) report that those machine learning methods cannot beat the simple logit model in expanding out-of-sample prediction for the banking crisis. A straightforward measure can prevail and Wolff and Neugebauer (2019) document that those machine learning methods cannot beat the HA benchmark. In comparison with previous studies, our conclusion is based on a more comprehensive survey of machine learning methods with a longer date period, more machine learning methods, and popular evaluation criteria. We confirm the value of the HA for equity return prediction task and report that complex machine learning methods do not necessarily guarantee more accurate forecasts. The complexity of those methods can mostly breed more complications and cost more computational effort. Complicated machine learning methods can at best exercise their superior prediction performance only when that multivariate dataset is sufficiently large and complex. Otherwise, we confirm that the HA is a straightforward and superior alternative.

The remainder of the paper proceeds as follows. Section 2 demonstrates the methodology of various machine learning models and their combination method. Section 3 outlines the dataset, including macroeconomic variables and technical indicators. Section 4 summarizes both the in-sample and out-of-sample prediction results and variable importance analysis. Section 5 investigates the economic value of machine learning forecasts with two benchmark strategies from a market timing perspective. Section 6 presents the robustness checks. Finally, Section 7 concludes.

2. Methodology

This section introduces the selected machine learning methods and evaluation criteria. We first consider a general task to forecast the log equity premium at the month $t+1$ given information available at month t, which can be expressed as:

$$r_{t+1} = E_t(r_{t+1}) + \epsilon_{t+1},\ t = 0, 1, \ldots, T-1,$$
$$\textit{where}\ E_t(r_{t+1}) = g(X_t),\ t = 0, 1, \ldots, T-1.$$

The log equity risk premium r_{t+1} is the monthly return on a stock market index over the risk-free interest rate, $E_t(r_{t+1})$ is the expected return at the time t, and the functional form of $g(.)$ is unspecified, which represents a machine learning model. X_t is a vector of explanatory variables, including macroeconomic variables and technical indicators, and ϵ_{t+1} denotes a zero-mean disturbance term at month $t+1$.

2.1. Forecasting models for the equity premium prediction

We evaluate seventeen models in total for predicting the equity premium, including the standard multivariate linear regression (OLS), fifteen machine learning methods, and a combination model that averages these sixteen models. The fifteen machine learning methods consist of eleven models used by Gu *et al.* (2020) and four additional popular models at our selection.

2.1.1. Machine learning models used in Gu *et al.* (2020). We follow Gu *et al.* (2020) and adopt their eleven selected machine learning methods: partial least squares (PLS), principal components regression (PCR), least absolute shrinkage and selection operator (LASSO), elastic net (ENet), gradient boosted regression trees (GBRT), random forest (RF), and neural networks with depth of hidden layers ranging from 1 to 5 (NN1 to NN5). We briefly overview these eleven machine learning models in Appendix A.

2.1.2. Four other popular machine learning methods. We also enrich the machine learning algorithms by adding four other competitors: Ridge, SVR, KNN, and XGBoost. We introduce these models as follows.

(1) Ridge
Ridge is a linear regression technique designed to address multicollinearity by adding a penalty term to the OLS regression (Hoerl and Kennard 1970). This penalty is the sum of the squares of the coefficients multiplied by a regularization parameter, which helps to shrink large coefficients towards zero, thereby reducing model complexity and improving generalization. The key benefit of ridge regression is its ability to handle multicollinearity and prevent overfitting, leading to more robust predictions on unseen data. Unlike LASSO regression, Ridge does not perform feature selection since it shrinks coefficients but does not set them to zero.

(2) SVR
SVR is one of the most popular machine learning methods for efficiently detecting non-linear relations (Cortes and Vapnik 1995). The idea behind SVR is to find a hyperplane that can separate observations into distinct groups. Mathematically, the hyperplane can maximize the distance of close observations. When observations are linearly separable, the hyperplane can be expressed by a small number of representative data points. For non-linear separable datasets, SVR uses a non-linear kernel mapping procedure to project data into a high-dimensional feature space. The advantage of the mapping procedure is that it can enlarge the space of function that is used to describe observations.

(3) KNN
KNN is a useful machine learning technique for both classification and regression (Fix and Hodges 1989, Altman 1992). KNN aims to find the k closest observations in the training set to make predictions. For the classification task, KNN outputs a class membership by a majority vote of its nearest neighbors. For the regression task, the output is the average value of its k nearest neighbors. The hyperparameter k is usually determined in the validation set. Although KNN is extensively used to predict financial crises (Holopainen and Sarlin 2017, Beutel *et al.* 2019), there is little attention to its application in forecasting stock returns.

(4) XGBoost
XGBoost (Chen and He 2015) is another popular tree-based boosted algorithm. XGBoost has become well-known to data scientists in the machine learning circle because it has won several machine learning competitions on Kaggle.[†] Like RF and GBRT, XGBoost also combines many weak learners to generate a boosted complex classifier. XGBoost provides an efficient, scalable implementation of the gradient boosting framework developed by Friedman (2001). A key feature of this algorithm is that it considers second-order Taylor approximation in the loss function to make a connection to the Newton–Raphson method. XGBoost is widely used to predict financial distress (Climent *et al*. 2019, Olson *et al*. 2021).

2.1.3. Forecast combination (Combination). The forecast combination method is confirmed to deliver strong predictive power in several previous studies (Rapach *et al*. 2010, Gu *et al*. 2020). The idea of this method is to reduce model uncertainty and instability of individual predictive models by combining the outcomes of these individual predictive models. We consider the average of various models to form the combination forecast, that is given as follows: $\hat{r}_{c,t+1} = \frac{1}{N}\sum_{i=1}^{N}\hat{r}_{i,t+1}$, where $\hat{c}_{i,t+1}$ denotes the combination forecast at month $t+1$, $\hat{r}_{i,t+1}$ is the individual forecast at month $t+1$, and N is the number of individual forecasts. In this study, the combination forecast is the average of the aforementioned sixteen forecasts (fifteen machine learning forecasts plus the OLS forecast).

2.2. *Sample splitting and hyperparameter tuning*

Tuning hyperparameters is of great importance when using machine learning methods for financial predictions. We follow the common machine learning practice (Gu *et al*. 2020, Bianchi *et al*. 2021) and split the data into training, validation, and testing set. The training and validation set is fixed at 85% and 15% of the in-sample data, respectively. To preserve the temporal ordering of the data, we avoid cross-validation by randomly selecting independent subsets of the in-sample data. Alternatively, our study uses a consequential training and validation set with an expanding window scheme. This scheme includes all the available historical observations and expands the data period whenever a new observation is added, to train the machine learning models. Figure 1 illustrates the recursive time-ordered validation scheme (Kelly and Xiu 2023).

To get the optimal hyperparameters, we first estimate the machine learning models with the mean square error as the loss function for the training set. We then use the validation set to iteratively search the hyperparameters that optimize the scoring function, a process often referred to as grid search. In our study, the default scoring function is also the mean square error function.[‡] Due to the computational intensity of machine learning algorithms, we avoid refitting models monthly. Instead, we choose to refresh our machine learning models annually, with updates occurring every 12 months. Appendix B provides the details of hyperparameter settings and their potential values for each model. After identifying the best hyperparameters for each machine learning model, we retrain the final models using the entire training set (including both training and validation data). These models are then applied to the test set, which is not used during the training phrase.

2.3. *Forecast evaluation*

Welch and Goyal (2008) report that if a model has no in-sample performance, its out-of-sample performance is not interesting. We follow their study and conduct both in-sample and out-of-sample tests in our study.

For the in-sample analyses, we employ two evaluation criteria to assess their forecasting power with the HA benchmark: the in-sample R-squared (R_{IS}^2) and the in-sample success ratio ($Succ_{IS}$). The HA benchmark is a stringent benchmark (Goyal and Welch 2003, Welch and Goyal 2008), which is given by $r_{B,t} = \frac{1}{t}\sum_{i=1}^{t} r_i$, where r_i represents the actual log equity premium. The R_{IS}^2 is defined as $R_{IS}^2 = 1 - \frac{MSPE_M}{MSPE_B}$, where $MSPE_M = \frac{1}{T}\sum_{i=1}^{T}(r_i - \hat{r}_i)^2, MSPE_B = \frac{1}{T}\sum_{i=1}^{T}(r_i - \hat{r}_{B,i})^2$. $MSPE_M$ denotes the mean squared prediction error (MSPE) of the forecasting model of interest while $MSPE_B$ is the MSPE of the benchmark model. $\hat{r}_{B,i}$ represents the estimated equity premium predicted by HA benchmark. $\hat{r}_i$ is an in-sample individual machine learning forecast or an in-sample combination forecast mentioned in Section 2.1. T is the length of full sample period. The R_{IS}^2 is a pointwise prediction measure that evaluates how much the forecasting model improves the MSPE as compared with the historical mean benchmark. Thus, when $R_{IS}^2 > 0$, the $\hat{r}_i$ forecast outperforms the benchmark model and vice versa.

The second criterion $Succ_{IS}$, also known as directional accuracy or hit ratio, is computes as $Succ_{IS} = \frac{1}{T}\sum_{\mathrm{t}=1}^{T} I(\hat{r}_i, r_i)$, where $I(\hat{r}_i, r_i)$ is an indicator function. The indicator function takes the value of 1 if both $\hat{r}_i$ and r_i are positive; otherwise, it returns 0. The success ratio measures the proportion of times the predicted direction of change (up or down) matches the actual direction of change in the equity premium. In other words, this criterion assesses the ability of a forecasting model to correctly predict the direction of whether the log equity premium will rise or fall. We make this additional attempt for the consideration that an investor's success (profit) has a stronger relationship with directional accuracy than point accuracy measured by in-sample R-squared (Leitch and Tanner 1991).

We do not include the coefficient of determination, widely used in previous literature, as our in-sample evaluation criterion (Welch and Goyal 2008, Rapach *et al*. 2010, Neely *et al*. 2014). It is noteworthy that the coefficient of

[†] https://www.kaggle.com/

[‡] The mean square error scoring function yields the same results as those obtained when using the coefficient of determination for selecting hyperparameters. The coefficient of determination, often denoted as R^2, is computed as 1-SSE/SST. Here, SSE is the sum of the squared differences between the actual equity premiums and the equity premiums predicted by the forecasting model. SST is the sum of squared differences between the actual equity premiums and the mean value of all the actual equity premiums. For a given validation set, SST remains fixed, whereas SSE corresponds to the mean square error scoring function. According to Campbell and Thompson (2008), a monthly R^2 of about 0.5% can represent economically significant degree of equity premium predictability for investors.

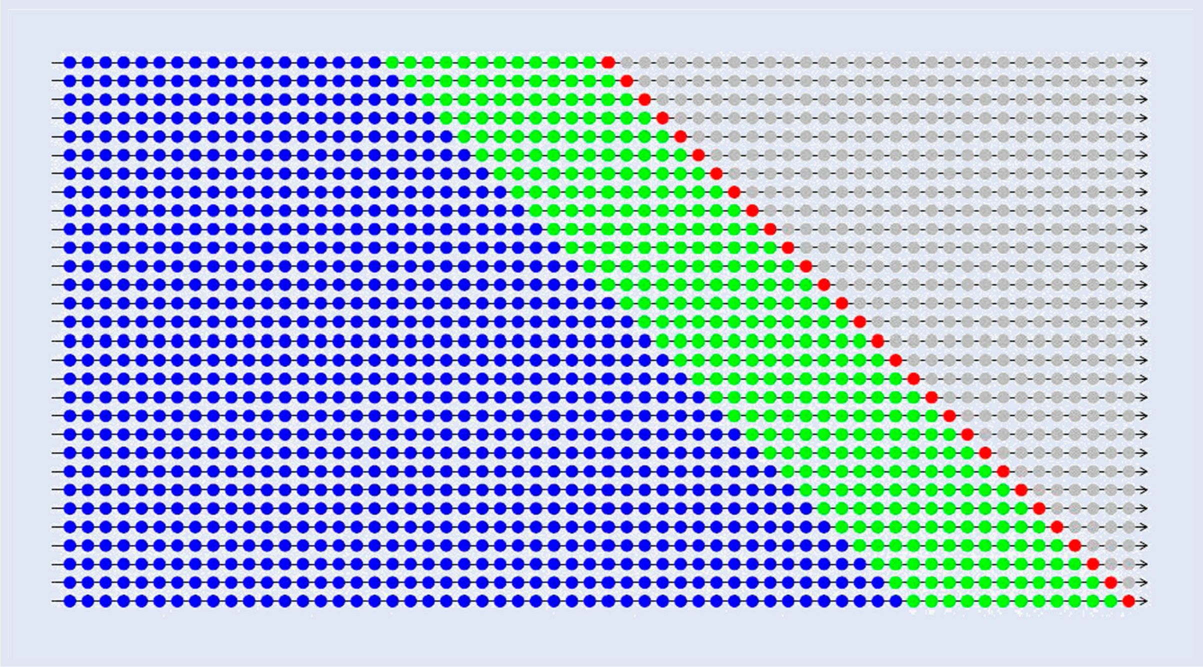

Figure 1. Illustration of Recursive Time-ordered Validation Scheme (Kelly and Xiu 2023). Notes: Blue dots indicate training observations, green dots indicate validation observations, and red dots indicate test observations. The training-to-validation ratio is 85:15. Each row represents a step in the recursive design.

determination is a pointwise prediction measure suitable for linear regression models, whereas most of our models are nonlinear. Besides, the evaluation criterion does not allow for a straightforward comparison with the HA benchmark but the in-sample R-squared does.

Furthermore, we include the out-of-sample R-squared (R^2_{OS}) and the out-of-sample success ratio ($Succ_{OS}$) for the out-of-sample tests. The R^2_{OS} statistic is akin to the R^2_{IS} statistic and is given as $R^2_{OS} = 1 - \frac{MSPE_M}{MSPE_B}$, where $MSPE_M = \frac{1}{q}\sum_{i=1}^{q}(r_{m+i} - \hat{r}_{m+i})^2, MSPE_B = \frac{1}{q}\sum_{i=1}^{q}(r_{m+i} - \hat{r}_{B,m+i})^2$. $\hat{r}_{B,m+i}$ represents the equity premium predicted by the HA benchmark in month $m + i$. m and q denote the lengths of in-sample period and out-of-sample period, respectively. We further employ the $MSFE - adjusted$ statistic by Clark and West (2007) to test the null hypothesis that $R^2_{OS} \leq 0$. The second criterion $Succ_{OS}$ is given by $Succ_{OS} = \frac{1}{q}\sum_{\mathrm{t}=1}^{q} I(\hat{r}_{m+i}, r_{m+i})$. We conduct the accuracy test of Pesaran and Timmermann (1992) to compare the predictive power of directional accuracy of machine learning methods with that of the historical mean. The null hypothesis is that the success ratio of the forecasting model of interest is less or equal to the success ratio of the random walk.

In addition, we follow Neely *et al.* (2014) and assess the comparative effectiveness of predicting equity risk premium during National Bureau of Economic Research (NBER)-dated business-cycle expansions (EXP) and recessions (REC). We thus compute the following out-of-sample R-squared statistics for out-of-sample evaluation: $R^2_{\mathrm{OS,c}} = 1 - \frac{\sum_{i=1}^{q} I^c_{m+i}(r_{m+i}-\hat{r}_{m+i})^2}{\sum_{i=1}^{q} I^c_{m+i}(r_{m+i}-\hat{r}_{B,m+i})^2}, c = \mathrm{EXP, REC}$, where $I^{EXP}_{m+i}(I^{REC}_{m+i})$ is an indicator function that takes 1 when $m + i$ is in an expansion period, and 0 otherwise.

3. Data

We employ two categories of good predictors of equity premium: macroeconomic predictors and technical predictors. To make our research comparable to previous studies (Welch and Goyal 2008), we focus on the updated data over the period 1926:12–2020:12. The sample period covers most of both the expansions and the recessions of the postwar era. The equity premium is measured as the difference between the log return of the S&P 500 equity market index (including dividends) and the log return on the risk-free Treasury bill. We respectively describe macroeconomic variables and technical indicators as follows.

3.1. Macroeconomic variables

In line with Elliott *et al.* (2013), we use twelve popular macroeconomic variables suggested by Welch and Goyal (2008). We exclude the long-term yield and the log dividend-earnings ratio to avoid the multicollinearity problem. These data are available on the personal website of Amit Goyal.† The twelve variables are:

1 Log dividend yield (DY): log of a 12-month moving sum of earnings on the S&P 500 index minus the log of lagged stock prices (S&P 500 index).
2 Log dividend-price ratio (DP): the difference between the log of dividends paid on the S&P 500 index and the log of stock prices.
3 Log earning-price ratio (EP): the log of earnings on the S&P 500 index minus the log of stock prices.

† https://sites.google.com/view/agoyal145/?redirpath=/

4 Stock return variance (SVAR): the monthly sum of squared daily returns on the S&P 500 index.

5 Book-to-market ratio (BM): the ratio of book value to market value for the Dow Jones Industrial Average.

6 Net equity expansion (NTIS): the ratio of a 12-month moving sum of net issues by New York Stock Exchange-listed stocks to the total end-of-year market capitalization of the New York Stock Exchange stocks.

7 Treasury bill rate (TBL): the interest rate on a three-month Treasury bill (secondary market).

8 Long-tern return (LTR): returns on long-term government bonds.

9 Term spread (TMS): the difference between the long-term yield and the Treasury bill rate.

10 Default yield spread (DFY): the difference between BAA- and AAA-rated corporate bond yields.

11 Default return spread (DFR): the difference between the long-term corporate bond returns and the long-term government bond returns.

12 Inflation (INFL): calculated from the Consumer Price Index (all urban consumers).

3.2. Technical indicators

Neely *et al.* (2014) document that technical indicators can provide complementary information to forecast the equity premium. We thus employ twelve technical predictors based on two trend-following strategies, the moving-average (MA) rule, and the momentum (MOM) rule. We do not consider the volume-based strategy in this study because the volume data of the S&P 500 from 1927 to 1956 are not available.

The MA rule will yield a buy or sell signal ($S_{i,t} = 1$ or $S_{i,t} = 0$, respectively) at the end of the month t by comparing two moving averages, $S_{i,t} = \begin{cases} 1 \text{ if } MA_{s,t} \geq MA_{l,t} \\ 0 \text{ if } MA_{s,t} < MA_{l,t} \end{cases}$, where $MA_{j,t} = \frac{1}{j}\sum_{i=0}^{j-1} P_{t-i}$, P_t denotes the level of the S&P 500 index, and $s(l)$ represents the length of the short (long) MA ($s < l$). The MA rule can capture the trend of the stock price as the short-term MA is more susceptible to recent price variation than the long-term MA. We consider the MA rules with $s = 1, 2, 3$ and $l = 9, 12$. Thus, the MA rules can yield six MA predictors (3∗2).

The MOM rule will generate a trading signal by comparing the stock index with its level of m months ago, $S_{i,t} = \begin{cases} 1 \text{ if } \mathrm{P}_t \geq P_{t-m} \\ 0 \text{ if } \mathrm{P}_t < P_{t-m} \end{cases}$. If the current stock index P_t is higher than its level $m$months ago, it suggests 'positive' momentum and provides a signal for buy-in. We consider $m = 1, 2, 3, 6, 9$ and 12 in this study and accordingly we have six momentum predictors.

4. Empirical results and discussions

This section demonstrates the in-sample and out-of-sample empirical results of applying the selected machine learning methods in Section 2.1 to predict the equity premium. We use seventeen competing models for empirical analyses: an OLS model, fifteen machine learning methods, and a combination approach averaging these sixteen models. We also include the HA forecast as the benchmark. Section 4.1 focuses on in-sample predictive performance for the entire sample period and Section 4.2 demonstrates the details of out-of-sample results. We also employ the variable importance technique, based on the reduction in out-of-sample R-squared, to identify the key predictors for predicting the equity premium.[†]

Table 1. In-sample performance for the entire sample period.

Forecasting models	In-sample R^2(%)	Success ratio (%)
OLS	5.51	60.51
PLS	2.22	60.07
PCR	2.05	59.09
LASSO	1.97	60.25
ENet	1.68	60.43
GBRT	10.73	60.96
RF	9.77	60.69
NN1	1.39	60.43
NN2	0.33	60.25
NN3	0.61	60.07
NN4	1.53	59.72
NN5	0.44	60.07
Ridge	2.65	60.25
SVR	– 4.77	42.15
KNR	15.38	66.1
XGBoost	45.06	75.51
Combination	8.99	64.42
HA	0	59.01

Notes: This table reports the in-sample forecasting performance for the entire sample period. The 'HA' in the first column and last row denotes the historical average benchmark. The in-sample R-squared in the second column measures the percent reduction in mean squared forecast error for the predictive model given in the first column relative to the HA forecast. The success ratio in the third column evaluates how well a forecasting model predicts whether the log equity premium will increase or decrease. ∗∗∗, ∗∗, and ∗ denote significance at 1%, 5%, and 10% levels, respectively. The entire sample period ranges from 1926:12–2020:12.

4.1. In-sample analyses

For in-sample analyses, we use the entire sample as the training set. We employ the method outlined in Section 2.2 to fine-tune the hyperparameters described in the Appendix B. Once we determine the optimal hyperparameters, we train the machine learning models on the full data set and then refit all points in the entire sample using these trained models. Table 1 reports the in-sample predictive performance of various machine learning models.

The results indicate that most forecasting models outperform the HA benchmark in terms of in-sample predictive performance for the entire sample period. The XGBoost model exhibits the best prediction performance with an impressive in-sample R-squared of 45.06% and a success ratio of 75.51%. Following XGBoost, KNR also performs well with an R-square of 15.38% and a success ratio of 66.1%. The other three models (GBRT, Combination, and RF) also show solid

[†] The replication data and code are available on GitHub: https://github.com/XingfuXu/EquityPremiumPrediction-Jupyter.

Table 2. Out-of-sample forecasting results.

Forecasting models	$R^2_{OS}(\%)$	CW-stat	Success ratio (%)	PT-stat	$R^2_{OS,REC}(\%)$	$R^2_{OS,EXP}(\%)$
OLS	− 12.68	0.61	56.06**	2.2	− 21.13	− 9.86
PLS	− 5.19	0.6	55.54*	1.33	− 7.18	− 4.52
PCR	0.23**	1.84	59.19***	3.02	2.44	− 0.51
LASSO	− 1.37	0.8	58.67**	1.88	− 4.65	− 0.28
ENet	− 1.42	0.75	58.54**	2.07	− 4.38	− 0.42
GBRT	− 14.45	1.13	58.02	0.6	− 17.62	− 13.4
RF	− 7.17	0.13	58.15	0.76	− 8.42	− 6.75
NN1	− 12.88	− 1.37	58.28	0.92	− 21.21	− 10.1
NN2	− 5.02	− 0.28	58.54	1.17	− 8.08	− 4
NN3	− 17.33	1.06	54.5	− 0.49	− 23.42	− 15.3
NN4	− 10.85	0.16	56.98	0.12	− 1.62	− 13.93
NN5	− 35.77	− 0.88	58.93	0.96	− 13.26	− 43.29
Ridge	− 1.18	0.92	58.54*	1.6	− 2.93	− 0.6
SVR	− 41.6	− 0.19	41.72	− 1.09	− 49.8	− 38.86
KNR	− 17.19	0.41	54.37*	1.49	− 19.11	− 16.55
XGBoost	− 34.95	0.64	55.41	0.8	− 38.34	− 33.82
Combination	− 1.58	0.61	58.02*	1.4	− 5.74	− 0.19
HA	0	–	59.97	–	0	0

Note: This table reports the out-of-sample forecasting performance using an expanding estimation window. The ‘HA’ in the first column and last row denotes the historical average benchmark. The out-of-sample R-squared in the second column measures the percent reduction in mean squared forecast error for the predictive model given in the first column relative to the HA forecast. The success ratio in the fourth column evaluates how well a model can predict whether the log equity premium will increase or decrease. The ‘CW-stat’ in the third column and ‘PT-stat’ in the fifth column corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). The last two columns display the out-of-sample R-squared over NBER-dated recessions and expansions. ‘-’ means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1957:01–2020:12.

performance with the all three achieving in-sample R-squared over 5% and success ratios above 60%.

The simple OLS model outperforms most other selected models, including PLS, PCR, LASSO, ENet, Ridge, and various neural networks (NN1 to NN5), which exhibit moderate performance improvements over the HA benchmark. OLS forecast demonstrates considerable predictive power with an R-square of 5.51% and a success ratio of 60.51%. SVR is the only model that underperforms the HA benchmark, with a negative R-square of -4.77% and a lower success ratio of 42.15%.

Overall, the findings suggest that while most machine learning models can beat the HA benchmark for the in-sample analyses, the effectiveness varies, with tree-based models (XGBoost, GBDT, and RF) leading the performance metrics. However, the good in-sample fit raises concerns about potential overfitting and data mining issues. (Inoue and Kilian 2005, Timmermann 2018). A good in-sample fit does not necessarily guarantee out-of-sample performance. We need out-of-sample diagnostics to further assess their forecasting ability.

4.2. Out-of-sample analyses

We conduct out-of-sample tests to investigate whether the excellent in-sample forecasts are stable and correctly specified for out-of-sample performance.

4.2.1. Out-of-sample performance. The out-of-sample estimation period starts from January 1957, which is in line with Neely *et al*. (2014). Table 2 exhibits the out-of-sample predictive results using the expanding window estimation scheme.

Based on the out-of-sample R-squared statistics presented in Table 2, various sophisticated machine learning models failed to surpass the HA benchmark. Specifically, the simple OLS forecast exhibits a poor out-of-sample R-squared of -12.68% as compared with the HA forecast but still outperforms seven machine learning models (GBRT, NN1, NN3, NN5, SVR, KNR, and XGBoost). This primary finding indicates that basic linear models can outperform machine learning models for out-of-sample prediction tasks. Among adopted machine learning methods, the five dimensionality reduction techniques (PLS, PCR, ENet, LASSO, and Ridge) demonstrate competitive performance as compared with other machine learning models. The PCR forecast is the only one that surpasses the HA forecast with a significant out-of-sample R-squared of 0.23%. The other four models exhibit less satisfactory performance, and their out-of-sample R-squared values range from -5.19% to -1.18%. The five neural network models (NN1-NN5) report out-of-sample R-squared varying from -35.77% to -5.02%, suggesting that they are not well-suited for this prediction task. Among the remaining four machine learning methods (RF, SVR, KNR, and XGBoost), RF performs the best with an out-of-sample R-squared of -7.17%, while XGBoost performs the worst with an out-of-sample R-squared of -34.95%. Finally, the combination approach shows relatively good performance with an out-of-sample R-squared of -1.58% compared to individual machine learning models.

It is worth noting that success ratios in Table 2 refer the directional prediction accuracy and contrast our model prediction outcome with the random walk forecast, rather than the HA benchmark (Pesaran and Timmermann 1992). Eight out of the seventeen forecasting models (OLS, PLS, PCR, LASSO, ENet, Ridge, KNR, and Combination) demonstrate significant directional accuracy, suggesting that they can outperform a random walk forecast when predicting the equity premium. Similarly to the performance of out-of-sample R-squared, the five dimensionality reduction techniques continue to outshine other machine learning models in terms of significant success ratios. For instance, PCR stands out for its ability to detect market movements of the equity premium, showing the highest success ratio of 59.19% among all seventeen competing models. By contrast, the success ratios of the five neural network models do not reach significance, ranging from 54.5% to 58.98%. The SVR model exhibits the lowest success ratio of 41.72%. Particularly, we find that the HA forecast even achieves a success ratio of 59.97%, slightly surpassing the best PCR forecast by 0.78%. This indicates that none of the seventeen competing models surpass the HA forecast in terms of success ratio.

The last two columns of Table 2 summarize out-of-sample R-squared for NBER-dated recessions and expansions. Most machine learning models exhibit higher out-of-sample R-squared during expansions than during recessions. This observation aligns with the understanding that stock market volatility level tends to be higher during recessions, making predictions more challenging (Schwert 1989, Hamilton and Lin 1996). PCR is the only exception, achieving a positive out-of-sample R-squared of 2.44% during recessions. This observation coincides with Neely *et al*. (2014) that PCR effectively captures the changing patterns of equity premiums in turbulent market conditions.

We further present the time-series plots of the difference between the cumulative square prediction error of the HA benchmark model and that of machine learning models in Figure 2. We focus on the forecasting performance of eight machine learning methods (PLS, PCR, LASSO, ENet, NN3, RN, NN2, NN4, and Ridge) which exhibit stronger predictive power than the other machine learning models in Table 2. An ascending curve in each panel indicates the machine learning model outperforms the HA, while a descending suggests underperformance. For any given out-of-sample period, if the jagged solid line at the end of the period is above the horizontal zero line, it means that the machine learning model's MSPE is lower than that of the HA over that period. This observation reflects that machine learning method outperforms the HA benchmark.

Furthermore, Figure 2 illustrates that none of the eight machine learning models consistently outperforms the historical mean benchmark in terms of the MSPE criterion. Specifically, the jagged solid lines of four methods (PLS, RF, NN2, and NN4) always lie below the horizontal zero line, indicating that they consistently underperform the HA benchmark, regardless of the chosen out-of-sample period. LASSO, ENet, and Ridge show a similar pattern: they stay above the horizontal zero line before the 2008 Global Financial Crisis but drop sharply since the recession. Only the PCR model exhibits strong predictive power during the recession but shows a decreasing trend and is only slightly above zero at the end of 2020. In summary, Figure 2 reinforces the conclusion from Table 2 that it is highly challenging for the selected machine learning models to outperform the HA benchmark in forecasting the equity premium.

4.2.2. Using macroeconomic variables and technical indicators separately. Considering that macroeconomic variables and technical indicators capture different types of information relevant for forecasting the equity premium, we further follow Neely *et al.* (2014) by employing these two sets of predictors separately to generate out-of-sample machine learning forecasts. Table 3 compares the forecasting predictability of macroeconomic variables versus technical indicators. The results indicate that all machine learning models have out-of-sample R-squared(s) and success ratios below those of the HA benchmark, 0 and 59.97%, respectively. This demonstrates that none of the machine learning models outperform the HA benchmark, regardless of the predictor set used. However, machine learning models using technical indicators show stronger forecasting performance compared to those using macroeconomic variables alone, in terms of both out-of-sample R-squared and success ratio. Our comparison supports the conclusion by Neely *et al.* (2014) that technical indicators are more effective for forecasting the equity premium. Additionally, while the PCR model shows a positive out-of-sample R-squared of 0.23 when combining both predictor sets in Table 2, it has negative values when using macroeconomic variables or technical indicators individually in Table 3. This observation highlights the benefit of integrating both types of information for better equity premium forecasts.

4.2.3. Using a robust scoring function on the validation set. We used mean square error as the scoring function on the validation set in our previous analyses. However, financial returns typically exhibit a heavy-tailed distribution and demand more robust forecasting methods (Gu *et al*. 2020). To select hyperparameters that are robust to outliers, we further use the median absolute error (MAE) scoring function on the validation set (Hampel 1974). The MAE is calculated by taking the median of all absolute differences between the actual equity premiums and the predicted values.

Table 4 reports the forecasting performance using MAE scoring function on the validation set. In contrast with the results in Table 2, Ridge regression emerges as the best performing model, displaying a significant out-of-sample R-squared of 0.31 and a success ratio of 61.54%.

While PCR also shows a slight improvement with a positive out-of-sample R-squared of 0.07 and a significant success ratio of 59.71%, most models (LASSO, RF, GBRT, and others) fail to exceed the HA benchmark. In brief, most machine learning methods still underperform the HA forecast in terms of out-of-sample R-squared and success ratio, indicating persistent challenges in leveraging machine learning for equity premium predictions.

In summary, Tables 2–4 indicate that the employed machine learning methods underperform the HA benchmark for equity premium predictions with out-of-sample expanding window estimation scheme. Our results align with the conclusions of Welch and Goyal (2008), which suggest that sophisticated equity premium forecasts generally underperform compared to the historical mean. The strong performance of machine learning methods is primarily observed in in-sample analyses,

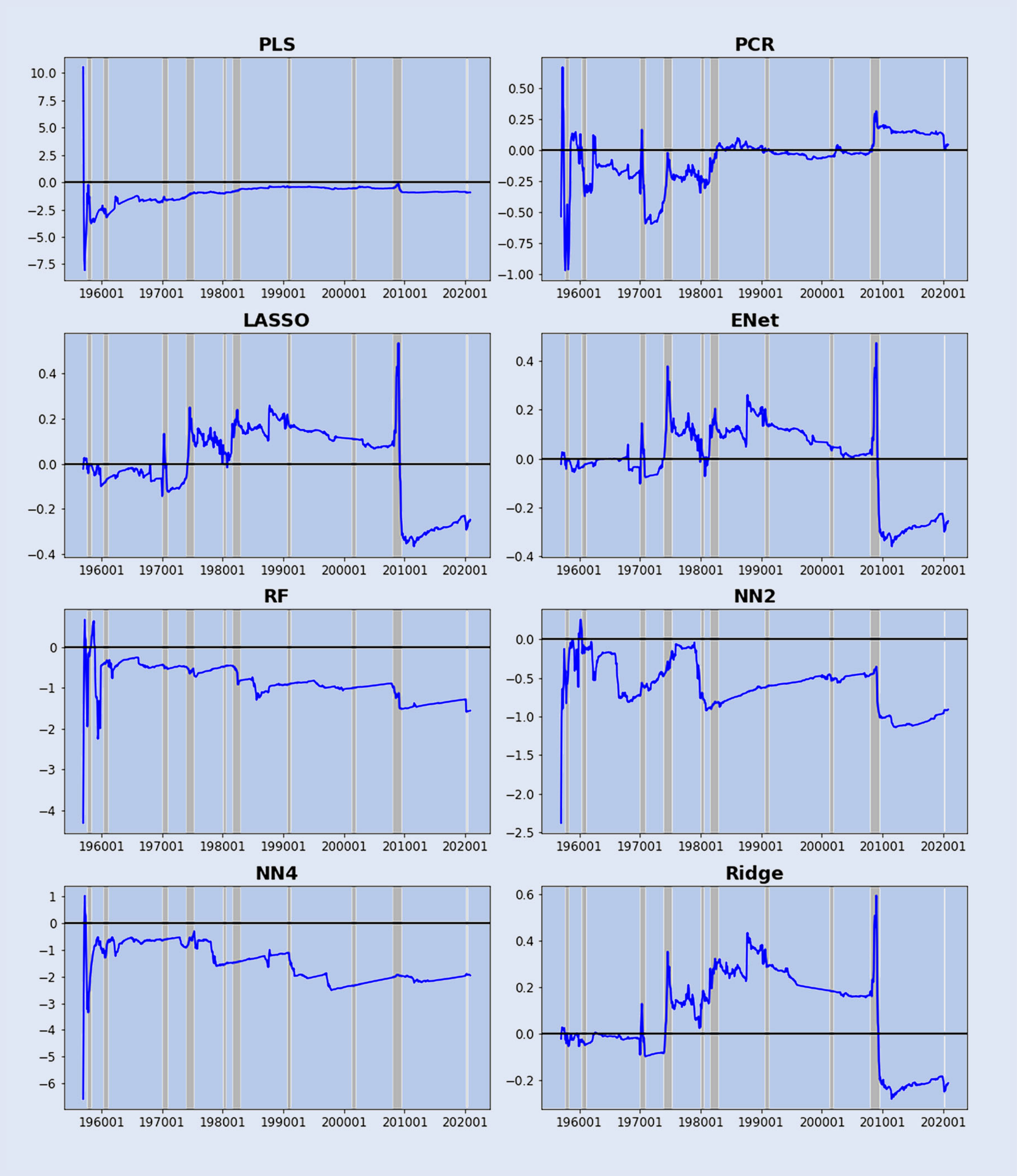

Figure 2. Cumulative square prediction error for the historical average benchmark model minus that of the machine learning models. Notes: The horizontal black solid line at y = 0 represents the performance level of the historical average benchmark model. In each panel, the rugged solid line shows the differences between the benchmark model and the machine learning models of interest; sections of this line above the horizontal black solid line of y = 0 indicate that the mean square prediction error of the machine learning model surpasses that of the historical mean model. Vertical shade bars represent NBER-dated recessions. The out-of-sample evaluation period ranges from 1957:01–2020:12.

but this does not necessarily translate to out-of-sample effectiveness. It is advisable not to be over-optimistic about the usage of machine learning in forecasting the equity premium. The excellent in-sample performance of machine learning can backfire in the context of real-world out-of-sample return predictions due to overfitting.

We notice two primary reasons that machine learning methods struggle with equity premium prediction. First, machine learning models typically require sufficiently large training datasets to perform effectively (Goldstein *et al.* 2021). Despite data period over 90 years, the monthly datasets used in this study are short by machine learning standards, which are designed to handle large-scale, complex, multi-dimensional data. Additionally, this study relies on empirical data rather than simulated data generated through experimentation. Second, the signal-to-noise ratio in return prediction is often low due to market efficiency and competitive pressures (Timmermann 2018). Predictors in this context are usually weak and

Table 3. Out-of-sample performance using macroeconomic variables and technical indicators separately.

	Macroeconomic variables				Technical indicators			
Forecasting models	R^2_{OS}(%)	CW-stat	Succes ratio (%)	PT-stat	R^2_{OS}(%)	CW-stat	Success ratio (%)	PT-stat
OLS	– 10.83	– 0.16	52.41	– 0.07	– 2.25	1.1	58.8**	2.07
PLS	– 4.21	– 0.47	54.63	0.02	– 1.33	0.89	59.19**	2.15
PCR	– 0.3*	1.46	57.24**	1.77	– 1.45	0.48	59.84***	2.72
LASSO	– 2.12	– 0.1	57.24	0.81	– 0.81	0.42	59.32	0.94
ENet	– 2.1	0.12	56.19	0.39	– 0.45	1.13	59.71**	1.74
GBRT	– 15.54	0.62	56.58	– 0.52	– 4.79	– 0.26	57.76	– 0.05
RF	– 10.94	– 0.48	54.76	– 1.84	– 2.76	0.47	57.76	1.28
NN1	– 4.8	1.02	59.06**	1.65	– 1.95	0.53	58.28	0.51
NN2	– 7.92	– 1.3	55.54	– 0.62	– 6.05**	1.78	57.11	0.67
NN3	– 13.84	0.28	59.06*	1.59	– 9.94	0.41	56.98	0.12
NN4	– 10.05*	1.44	58.67*	1.32	– 21.89	– 0.5	55.67	– 1.57
NN5	– 12.99	– 1.1	56.06	– 1.44	– 26.11	0.79	57.37	0.16
Ridge	– 2.29	– 0.05	57.5	0.8	– 0.8	0.85	58.15	0.33
SVR	– 24.37	0.47	41.72	– 1.79	– 19.63	0.73	46.15	0.95
KNR	– 18.16	– 0.18	51.76	– 0.59	– 15.64**	1.77	55.28	1.03
XGBoost	– 31.42	1.24	54.63	– 0.25	– 5.09	– 0.52	57.63	– 0.15
Combination	– 0.84	0.58	58.54	1.17	– 0.02*	1.42	58.67*	1.41
HA	0	–	59.97	–	0	–	59.97	–

Notes: This table reports the out-of-sample forecasting performance using macroeconomic variables and technical indicators separately. The 'HA' in the first column and last row denotes the historical average benchmark. The out-of-sample R-squared measures the percent reduction in mean squared forecast error for the predictive model given in the first column relative to the HA forecast. The success ratio evaluates how well a model can predict whether the log equity premium will increase or decrease. The 'CW-stat' and 'PT-stat' corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '-' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1957:01–2020:12.

low in predictive power. The adaptive nature and dynamic characteristics of markets further complicated the return prediction task (Israel *et al.* 2020).

In short, equity premium prediction presents a unique challenge characterized by small dataset size and low signal-to-noise ratios, hindering machine learning excelling. Meanwhile, our study reports that machine learning fails to outperform the HA benchmark for the equity premium predictions and our conclusion only holds for the given not sufficiently large dataset with the low signal-to-noise ratio. We cannot deny the noteworthy predictive power of machine learning methods in detecting relationships in large multi-dimensional datasets with complex structures. That is, we admit this limitation of our study, and we do not improve any employed machine learning method to enhance their predictive power for return predictions. We leave this improvement task for future research.

4.3. *Variable importance*

We proceed to use the variable importance technique to identify key predictors with high forecasting power in better-performing models. We follow the ranking approach in Gu *et al.* (2020) and measure variable importance of a specific variable j by the reduction in out-of-sample R-squared. Specifically, for a given forecasting model of interest, we first re-estimate the out-of-sample R-squared by setting all values of the predictor j to zero while keeping the other predictors unchanged during training. The reduction in out-of-sample R-squared is then calculated as the difference between the original out-of-sample R-squared using the full, unaltered predictors and the re-estimated out-of-sample R-squared using perturbed predictors.

We investigate the influential predictors for two categories of predictors (macroeconomic variables and technical indicators) based on their rankings of the reduction in out-of-sample R-squared. Figure 3 shows the variable importance of the top six predictors for the eight well-performing machine learning methods based on Table 2. A higher reduction indicates greater importance. For macroeconomic variables, the models agree on the most influential predictors: LTR (ranked first by one model and confirmed by six), TBL (ranked first by two models and confirmed by four), and TMS (ranked first by one model and confirmed by four). All these three variables are related to bond interest rates. Relating to the close relationship between equity premium and bond interest rate, the impact of LTR and TMS on the equity premium can be explained by the compensation hypothesis, which states that the equity premium compensates for exposure to discount-rate shocks affecting all long-term stocks (Fama and French 1989), while TBL can influence stock returns by reflecting changing uncertainty about inflation (Campbell 1987). For technical indicators, the rankings are more mixed, but different types of momentum indicators (MOM_1, MOM_2, MOM_3) are confirmed by four models. The effectiveness of momentum variables is attributed to the investor's underreaction to information (Hong and Stein 1999, Cohen and Frazzini 2008).

5. Economic value of market timing

The potential for a forecasting model to generate profit is of great interest to investors. To assess the economic value of the eight well-performing forecasts listed in Table 2 and the HA benchmark, we analyze their performance from a market

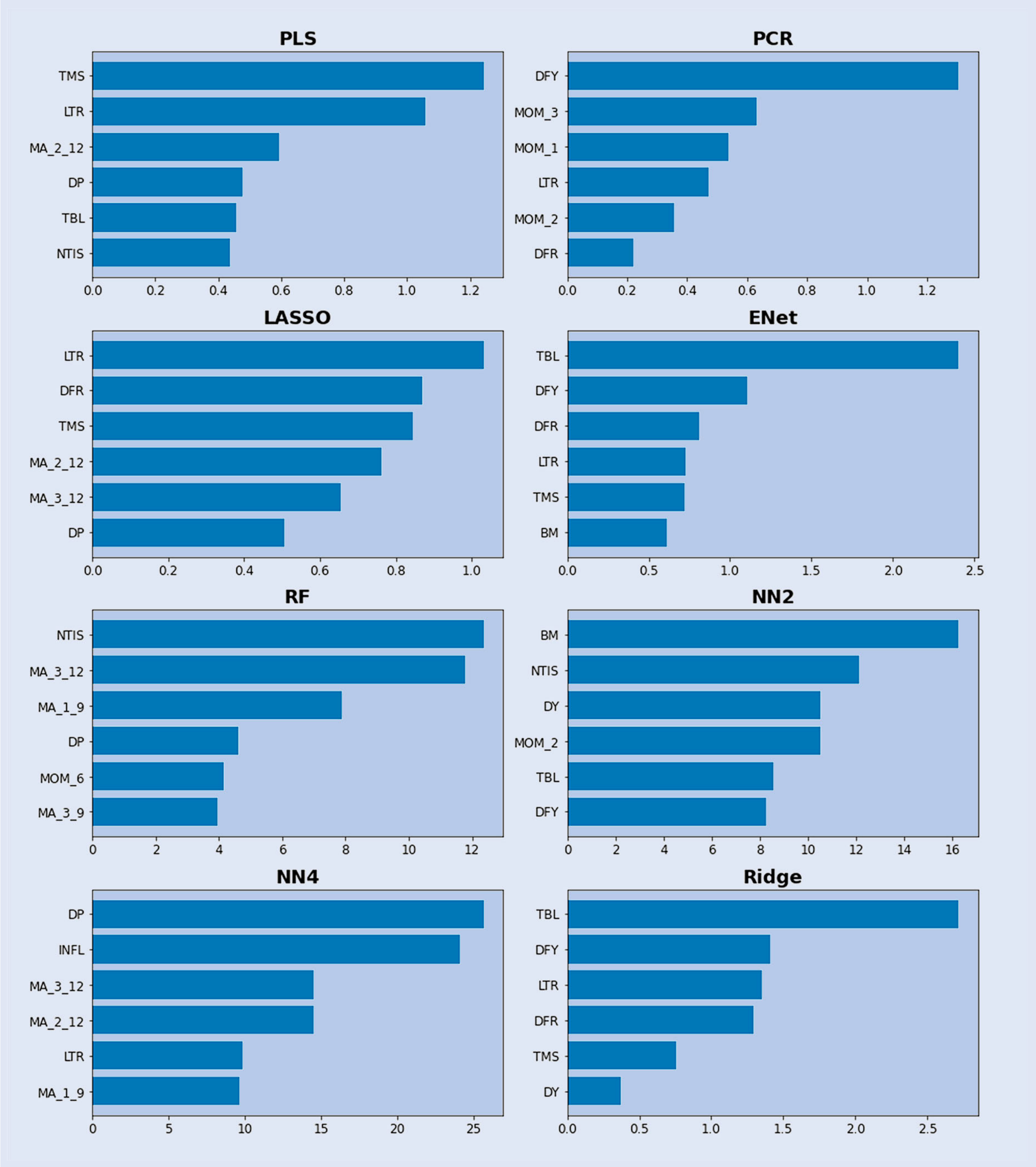

Figure 3. Variable importance of the eight selected machine learning models. Notes: This figure displays the top six influential predictors for the eight well-performing machine learning methods listed in Table 2. The out-of-sample evaluation period ranges from 1957:01–2020:12.

timing perspective. Market timing strategies rely solely on the success ratios of the corresponding forecasts (Gao *et al*. 2018, Wolff and Neugebauer 2019, Kelly *et al*. 2024).

The employed market timing strategies are based on the predicted log equity premium of different forecasts as follows. Each month t, we first transform the forecasting log equity premiums into simple equity premiums and use theses transformed values as timing signals to trade S&P 500 futures. Investors will take a long position in the S&P 500 futures at the beginning of month $t + 1$ if the predicted simple equity premium is positive or take a short position otherwise. These positions are closed at the end of month $t + 1$. To measure the performance of various market timing strategies, we calculate their annualized average returns and Sharpe ratios (computed as the average excess returns divided by their volatilities).

Table 5 summarizes the market timing performance of eight well-performing machine learning models with two benchmark strategies. The first benchmark strategy HA (the last row) is based on the predicted equity premium of the HA forecast, while the second buy-and-hold benchmark strategy (the first row) always takes a long position in the S&P 500. The buy-and-hold strategy achieves an average return of 11.15% with a Sharpe ratio of 0.14. All eight machine learning models outperform the buy-and-hold strategy in terms of both average return and Sharpe ratio. Among them, LASSO emerged as the top performer, achieving an average return of 22.41%

Table 4. Out-of-sample performance using the median absolute error scoring function on the validation set.

Forecasting models	$R^2_{OS}(\%)$	CW-stat	Success ratio (%)	PT-stat
OLS	– 12.68	0.61	56.06**	2.2
PLS	– 4.7	0.17	55.67	0.72
PCR	0.07**	1.93	59.71***	3.11
LASSO	– 0.7	1.2	58.8***	2.63
ENet	– 1.77	0.77	58.93***	2.65
GBRT	– 22.58	– 0.48	56.58	– 0.87
RF	– 16.51	0.34	56.19	– 0.97
NN1	– 26.87	– 1.03	58.41*	1.46
NN2	– 13.61**	1.33	57.11	0.63
NN3	– 20.5	– 0.11	55.93	0.41
NN4	– 16.44	– 0.45	55.54	– 0.75
NN5	– 29.21***	2.74	58.8**	1.99
Ridge	0.31*	1.47	61.54***	3.68
SVR	– 61.13	– 0.16	42.89	– 1.53
KNR	– 18.51	0.03	55.02*	1.62
XGBoost	– 32.89	0.76	54.11	– 0.17
Combination	– 1.01	1.03	58.67**	1.77
HA	0	–	59.97	–

Note: This table reports the out-of-sample forecasting performance using the median absolute error scoring function on the validation set. The 'HA' in the first column and last row denotes the historical average benchmark. The out-of-sample R-squared in the second column measures the percent reduction in mean squared forecast error for the predictive model given in the first column relative to the HA forecast. The success ratio in the fourth column evaluates how well a model can whether the log equity premium will increase or decrease. The 'CW-stat' in the third column and 'PT-stat' in the fifth column corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '–' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1957:01–2020:12.

Table 5. Market timing performance.

Market timing strategy	Average return (%)	Sharpe ratio
Buy-and-hold	11.15	0.14
PLS	14.68	0.39
PCR	19.03	0.52
LASSO	22.41	0.63
ENet	22.22	0.63
RF	21.39	0.63
NN2	21.98	0.63
NN4	19.04	0.54
Ridge	21.51	0.59
HA	25.63	0.8

Notes: This table reports the market timing results for the eight well-performing machine learning models listed in Table 2 and two benchmark strategies. 'HA' denotes the historical average benchmark strategy while the buy-and-hold strategy serves as another benchmark. The average return is annualized and presented as a percentage. The out-of-sample evaluation period ranges from 1957:01–2020:12.

and a Sharpe ratio of 0.63. However, the HA strategy demonstrates the best performance with the highest average return of 25.63% and the best Sharpe ratio of 0.8. Overall, the HA benchmark strategy delivers the highest economic value from a market timing perspective, making it the best choice for investors.

6. Robustness checks

We further test our conclusions with six variations in modeling settings. These variations include different rolling window estimation schemes, alternative forecasting window sizes, forecasting simple equity premium instead of log equity premium, alternative validation scheme for hyperparameter tuning, adding newly identified variables, and using alternative macroeconomic datasets. For brevity, we focus on the eight better-performing machine learning methods listed in Table 2: PLS, PCR, LASSO, ENet, RF, NN2, NN4, and Ridge.

6.1. *Rolling window estimation schemes*

The rolling window estimation scheme is another popular framework in out-of-sample return predictions (Campbell and Thompson 2008, Giovannelli *et al*. 2021). We fix the span of the rolling window as 5, 10, and 20 years. Based on the out-of-sample R-squared and success ratio estimates in Table 5, we conclude that the performance of the selected machine learning models improves as the length of the rolling window increases. However, all the employed machine learning models still show inferior predictive performance compared to the HA benchmark Table 6.

6.2. *Alternative forecasting window sizes*

Rossi and Inoue (2012) highlight that forecast window size affects out-of-sample performance. To minimize forecasting bias from window sizes, we choose to evaluate the most recent 30 years (1990:01–2020:12). To exclude the Oil Shock recession period of 1973–1975, Welch and Goyal (2008) claim that the prescience of many existing forecasts gains their predictive power from this period. Table 7 summarizes the out-of-sample forecasting results with alternative forecasting window sizes. PCR exhibits the highest out-of-sample R-squared (0.29%) and success ratio (60.92%) among the eight evaluated machine learning models, but the out-of-sample R-squared is not statistically significant and the success ratio remains lower than HA's 64.15% level.

6.3. *Forecasting simple equity premium*

In addition to the widely used log equity premium, we follow several previous studies and employ the simple equity premium as target variable (Kelly and Pruitt 2013, Martin 2017, Goyal *et al*. 2024). Table 8 reveals that most machine learning models fail to surpass the performance of the HA benchmark in both criteria. Only PCR significantly improves on HA in terms of out-of-sample R-squared (0.36%), while NN2 achieves a higher success ratio (60.63%) than HA but does not provide better out-of-sample R-squared. Overall, the HA benchmark continues to perform well.

6.4. *Alternative validation scheme for hyperparameter tuning*

This paper uses a sequential training and validation set for hyperparameter tuning to avoid inadvertent information

Table 6. Out-of-sample forecasting results with different rolling window estimation schemes.

	5-year rolling window		10-year rolling window		20-year rolling window	
Models	R^2_{OS}(%)	Success ratio (%)	R^2_{OS}(%)	Success ratio (%)	R^2_{OS}(%)	Success ratio (%)
PLS	– 47.24	52.41	– 15.89	55.41*	– 9.85	54.5*
	(1.08)	(– 0.16)	(1.05)	(1.36)	(0.16)	(1.52)
PCR	– 11.83	56.06	– 6.9	57.76**	– 4.37	58.54***
	(0.82)	(1.01)	(0.9)	(1.72)	(0.79)	(2.49)
LASSO	– 82.46	57.11	– 4.77**	59.84***	– 3.55	56.45
	(0)	(1.12)	(1.97)	(2.56)	(0.88)	(1.13)
ENet	– 74.4	55.41	– 5.35**	59.84***	– 4.27	57.89**
	(0.61)	(0.59)	(1.98)	(2.75)	(0.66)	(2.09)
RF	– 16.91	52.54	– 14.84	51.37	– 16.13	54.37
	0.19	(– 0.23)	0.53	(– 0.22)	(– 0.31)	0.12
NN2	– 102.98	46.68	– 100.03	52.54	– 30.5	56.06
	(– 0.3)	(– 2.47)	(0.71)	(0.23)	(0.5)	(1.1)
NN4	– 358.21	51.5	– 103.31	55.54	– 76.56	54.89
	(– 0.81)	(– 1.01)	(– 1.07)	(1.16)	(– 1)	(– 0.58)
Ridge	– 26.63	56.45	– 5.56	58.93**	– 2.59	58.02**
	(0.14)	(0.79)	(1.27)	(2.33)	(1.14)	(1.81)
HA	0	59.97	0	59.97	0	59.97
	(–)	(–)	(–)	(–)	(–)	(–)

Notes: This table reports the out-of-sample results using three rolling windows of fixed spans: 5, 10, and 20 years. 'HA' denotes the historical average benchmark. The statistics in parentheses corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '-' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1957:01–2020:12.

Table 7. Out-of-sample forecasting results using alternative forecasting window sizes.

Models	R^2_{OS}(%)	CW-stat	Success ratio (%)	PT-stat
PLS	– 8.45	– 0.15	54.45	0.09
PCR	0.29	0.99	60.92**	1.7
LASSO	– 4.16	0.12	60.65	– 0.04
ENet	– 4.17	0.06	60.11	– 0.28
RF	– 15.35	– 2.45	57.41	– 1.33
NN2	– 0.22	1.07	63.07	0.4
NN4	– 3.56	0.98	60.65	1.11
Ridge	– 4.7	– 0.03	60.65	0.23
HA	0	–	64.15	–

Notes: This table reports the out-of-sample forecasting performance in terms of out-of-sample R-squared and success ratio using alternative forecasting window sizes. 'HA' denotes the historical average benchmark. The 'CW-stat' and 'PT-stat' corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '-' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1990:01–2020:12.

Table 8. Out-of-sample forecasting results using the simple equity premium as the target variable.

Models	R^2_{OS}(%)	CW-stat	Success ratio (%)	PT-stat
PLS	– 5.6	0.62	54.63	0.56
PCR	0.36**	2.13	59.06**	2.33
LASSO	– 1.5	0.74	59.19**	2.07
ENet	– 1.46	0.81	59.97***	2.5
RF	– 8.84	– 1.09	55.67	– 1.46
NN2	– 7.09	1.06	60.63***	2.5
NN4	– 42.4	– 1.09	55.93	– 2.07
Ridge	– 1.1	1.03	59.71**	2.25
HA	0	–	59.97	–

Notes: This table reports the out-of-sample forecasting performance in terms of out-of-sample R-squared and success ratio using the simple equity premium as the target variable. 'HA' denotes the historical average benchmark. The 'CW-stat' and 'PT-stat' corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '-' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1957:01–2020:12.

leakage backward in time. However, maintaining the temporal order of the training and validation samples is not strictly necessary and may result in inefficient use of data for model selection (Kelly and Xiu 2023). To address this, we employ the widely used K-fold cross-validation scheme to select the optimal hyperparameters for each machine learning model. Table 9 shows the out-of-sample forecasting performance using a typical five-fold cross-validation scheme. We report that only LASSO beats on the HA forecast with a minor but not significant improvement of 0.2% in out-of-sample R-squared. None of the eight competing models outperform the HA forecast in terms of success ratio.

6.5. Adding newly identified variables

A host of papers claim that they beat the HA with some newly identified variables since Welch and Goyal (2008). We thus consider adding the following six new monthly time series to our predictors: variance premium (Bekaert and Hoerova 2014), variance risk premium (Bollerslev *et al.* 2009), changes in oil prices (Driesprong *et al.* 2008), distilled investor sentiment index (Huang *et al.* 2015), stock return dispersion (Maio 2016), and short interest (Rapach *et al.* 2016). Table 7 presents the results of incorporating these newly identified variables and report that all the machine learning

Table 9. Out-of-sample forecasting results using five-fold cross validation scheme.

Models	$R^2_{OS}(\%)$	CW-stat	Success ratio (%)	PT-stat
PLS	– 1.05**	1.89	59.32***	3.21
PCR	– 2.18	0.28	57.37*	1.61
LASSO	0.2	1.14	59.45***	1.86
ENet	– 0.17	0.94	58.54	1.2
RF	– 7.54	– 0.81	57.11	– 1.16
NN2	– 0.48	0.52	59.32	– 0.4
NN4	– 0.98	– 0.29	59.19	0.82
Ridge	– 0.41	0.81	58.93**	1.71
HA	0	–	59.97	–

Notes: This table reports the out-of-sample forecasting performance in terms of out-of-sample R-squared and success ratio using five-fold cross validation scheme. 'HA' denotes the historical average benchmark. The 'CW-stat' and 'PT-stat' corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '-' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1957:01–2020:12.

Table 10. Out-of-sample results with newly identified variables.

Models	$R^2_{OS}(\%)$	CW-stat	Success ratio (%)	PT-stat
PLS	– 17.1	– 0.54	60.16	1.08
PCR	– 8.04	– 2.22	56.57	– 1.66
LASSO	– 16.26	– 1.1	57.77	– 1.6
ENet	– 12.75	– 1.16	58.17	– 1.29
RF	– 13.31	– 0.8	57.37	0.08
NN2	– 23.35	– 0.18	57.37	0.74
NN4	– 83.78	1.27	54.98	0.03
Ridge	– 6.33	– 0.07	60.96	– 0.31
HA	0	–	62.95	–

Notes: This table reports the out-of-sample forecasting performance in terms of out-of-sample R-squared and success ratio with newly identified variables. 'HA' denotes the historical average benchmark. The 'CW-stat' and 'PT-stat' corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '-' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1990:01–2020:12.

Table 11. Out-of-sample results using alternative macroeconomic dataset.

Models	$R^2_{OS}(\%)$	CW-stat	Success ratio (%)	PT-stat
PLS	– 12.78	– 1.41	57.95	– 0.84
PCR	– 2.66	– 0.09	59.3	– 2.26
LASSO	– 3.82	– 2.14	57.41	– 2.17
ENet	– 3.53	– 0.94	57.41	– 1.44
RF	– 21.42	– 1.04	53.91	0.92
NN2	– 11.78	– 0.83	56.33	– 0.41
NN4	– 17.87	– 0.02	64.15	1.04
Ridge	0.12	1.18	63.61	0.67
HA	0	–	64.15	–

Notes: This table reports the out-of-sample forecasting performance in terms of out-of-sample R-squared and success ratio using alternative macroeconomic datasets. 'HA' denotes the historical average benchmark. The 'CW-stat' and 'PT-stat' corresponding to out-of-sample R-squared and success ratio are their respective test statistics based on Clark and West (2007) and Pesaran and Timmermann (1992). '-' means that the statistics are not available. $***$, $**$, and $*$ denote significance at 1%, 5%, and 10% levels, respectively. The out-of-sample evaluation period ranges from 1990:01–2020:12.

methods cannot display stronger predictive ability than the HA concerning both evaluation criteria. Indeed, Goyal *et al.* (2024) reconstruct more than 20 predictive variables from 26 recent studies on equity prediction. Despite their extensive survey, the authors document that only a small number of these variables continue to perform well both in-sample and out-of-sample. In short, adding new identified variables does not help enhance prediction performance Table 10.

6.6. Alternative macroeconomic datasets

McCracken and Ng (2016) provide a comprehensive macro database called FRED-MD to facilitate big-data macro finance research. This database covers a variety of economic categories, such as output and income, orders and inventories, interest and exchange rates, housing and consumption, and stock markets. We employ a total of 115 macroeconomic predictors without missing data from this database. The data period spans from 1959:01–2020:12. Table 11 reports the out-of-sample forecasting results for the most recent 30 years. The Ridge model displays a slightly higher but not significant out-of-sample R-squared of 0.12 in contrast to the HA forecast. The success ratio of the HA forecast remains the highest at 64.15%, though NN4 matches this value. These results confirm the robustness of our conclusions with this alternative macroeconomic dataset.

7. Conclusions

Machine learning methods have received considerable attention in their application in equity return predictions. We evaluate the efficacy of seventeen forecasting models, including the standard multivariate linear regression, fifteen machine learning methods, and a combination model for predicting the equity premium. We extensively examine the predictive power of the machine learning methods against the popular HA benchmark in both in-sample and out-of-sample analyses. These machine learning methods fail to maintain their outperformance in out-of-sample predictions though these methods, particularly tree-based models, demonstrate strong in-sample predictive power.

Regarding the HA benchmark outperformance in out-of-sample prediction, this conclusion remains valid even when we employ the macroeconomic variables and technical indicators separately or use an alternative scoring function on the validation set. Our empirical results support the conclusions of Welch and Goyal (2008) and confirm the excellent performance of the HA forecast. We attribute machine learning's underperformance to the small sample size and the low signal-to-noise ratio typical of return predictions.

We also conduct variable importance analysis to identify key predictors and adopt a ranking method to measure variable importance based on the reduction in out-of-sample R-squared when specific predictors are perturbed. We report that the major predictors for macroeconomic variables are LTR, TBL, and TMS, all related to bond interest rates. However, the results are inconclusive for technical indicators.

We further investigate the economic value of eight better-performing machine learning methods from a market timing perspective. We compare their timing performance with two benchmark strategies: the HA strategy and the buy-and-hold strategy. The results shows that all the eight machine learning models outperform the buy-and-hold strategy but fail to surpass the HA benchmark strategy in terms of average returns and Sharpe ratios.

Finally, we conduct the robustness checks across six settings: different rolling window sizes, alternative forecasting windows, the use of simple versus log equity premiums, the inclusion of newly identified predictive variables, and the use of alternative macroeconomic datasets. These results confirm our conclusions are robust and we cannot be over-optimistic about the performance of machine learning in predicting out-of-sample equity premiums.

Overall, this study contributes to enriching the literature on the long-standing debate of equity premium predictions by presenting a detailed investigation based on the selected machine learning methods to compare their predictive power with that of the historical mean. We highlight the persistent challenge of achieving robust out-of-sample performance in the context of equity premium forecasting and advocate for future research to improve existing machine learning algorithms to enhance their out-of-sample forecasting ability.

Open Scholarship

This article has earned the Center for Open Science badges for Open Data and Open Materials through Open Practices Disclosure. The data and materials are openly accessible at https://github.com/XingfuXu/EquityPremiumPrediction-Jupyter

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Wei-han Liu http://orcid.org/0000-0001-9077-4322

References

Akbari, A., Ng, L. and Solnik, B., Drivers of economic and financial integration: A machine learning approach. *J. Empir. Financ.*, 2021, **61**, 82–102.

Altman, N. S., An introduction to kernel and nearest-neighbor nonparametric regression. *Am. Stat.*, 1992, **46**(3), 175–185.

Bekaert, G. and Hoerova, M., The VIX, the variance premium and stock market volatility. *J. Econom.*, 2014, **183**(2), 181–192.

Beutel, J., List, S. and von Schweinitz, G., Does machine learning help us predict banking crises? *J. Financ. Stab.*, 2019, **45**, 100693.

Bianchi, D., Büchner, M., Tamoni, A. and Van Nieuwerburgh, S., Bond risk premiums with machine learning. *Rev. Financ. Stud.*, 2021, **34**(2), 1046–1089.

Bollerslev, T., Tauchen, G. and Zhou, H., Expected stock returns and variance risk premia. *Rev. Financ. Stud.*, 2009, **22**(11), 4463–4492.

Breiman, L., Random forests. *Mach. Learn.*, 2001, **45**(1), 5–32.

Campbell, J. Y., Stock returns and the term structure. *J. Financ. Econ.*, 1987, **18**(2), 373–399.

Campbell, J. Y. and Thompson, S. B., Predicting excess stock returns out of sample: Can anything beat the historical average? *Rev. Financ. Stud.*, 2008, **21**(4), 1509–1531.

Chen, T. and He, T., Higgs boson discovery with boosted trees. NIPS 2014 workshop on high-energy physics and machine learning, 2015.

Chinco, A., Clark-Joseph, A. D. and Ye, M. A. O., Sparse signals in the cross-section of returns. *J. Financ.*, 2018, **74**(1), 449–492.

Clark, T. E. and West, K. D., Approximately normal tests for equal predictive accuracy in nested models. *J. Econom.*, 2007, **138**(1), 291–311.

Climent, F., Momparler, A. and Carmona, P., Anticipating bank distress in the Eurozone: An extreme gradient boosting approach. *J. Bus. Res.*, 2019, **101**, 885–896.

Cohen, L. and Frazzini, A., Economic links and predictable returns. *J. Financ.*, 2008, **63**(4), 1977–2011.

Cortes, C. and Vapnik, V., Support-vector networks. *Mach. Learn.*, 1995, **20**(3), 273–297.

Cybenko, G., Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.*, 1989, **2**(4), 303–314.

Dangl, T. and Halling, M., Predictive regressions with time-varying coefficients. *J. Financ. Econ.*, 2012, **106**(1), 157–181.

Diaconis, P. and Shahshahani, M., On nonlinear functions of linear combinations. *SIAM J. Sci. Stat. Comput.*, 1984, **5**(1), 175–191.

Dichtl, H., Drobetz, W., Neuhierl, A. and Wendt, V.-S., Data snooping in equity premium prediction. *Int. J. Forecast.*, 2021, **37**(1), 72–94.

Driesprong, G., Jacobsen, B. and Maat, B., Striking oil: Another puzzle? *J. Financ. Econ.*, 2008, **89**(2), 307–327.

Elliott, G., Gargano, A. and Timmermann, A., Complete subset regressions. *J. Econom.*, 2013, **177**(2), 357–373.

Fama, E. F. and French, K. R., Business conditions and expected returns on stocks and bonds. *J. Financ. Econ.*, 1989, **25**(1), 23–49.

Feng, G., Giglio, S. and Xiu, D., Taming the factor zoo: A test of new factors. *J. Financ.*, 2020, **75**(3), 1327–1370.

Ferreira, M. A. and Santa-Clara, P., Forecasting stock market returns: The sum of the parts is more than the whole. *J. Financ. Econ.*, 2011, **100**(3), 514–537.

Fix, E. and Hodges, J. L., Discriminatory analysis. Nonparametric discrimination: Consistency properties. *Int. Stat. Rev./Revue Internationale de Statistique*, 1989, **57**(3), 238–247.

Freyberger, J., Neuhierl, A., Weber, M. and Karolyi, A., Dissecting characteristics nonparametrically. *R. Financ. Stud.*, 2020, **33**(5), 2326–2377.

Friedman, J. H., Greedy function approximation: A gradient boosting machine. *Ann. Stat.*, 2001, **29**(5), 1189–1232.

Gao, L., Han, Y., Li, S. Z. and Zhou, G., Market intraday momentum. *J. Financ. Econ.*, 2018, **129**(2), 394–414.

Giovannelli, A., Massacci, D. and Soccorsi, S., Forecasting stock returns with large dimensional factor models. *J. Empir. Financ.*, 2021, **63**, 252–269.

Goldstein, I., Spatt, C. S. and Ye, M., Big data in finance. *Rev. Financ. Stud.*, 2021, **34**(7), 3213–3225.

Goyal, A. and Welch, I., Predicting the equity premium with dividend ratios. *Manage. Sci.*, 2003, **49**(5), 639–654.

Goyal, A., Welch, I. and Zafirov, A., A comprehensive 2022 look at the empirical performance of equity premium prediction. *Rev. Financ. Stud.*, 2024, hhae044. DOI: 10.1093/rfs/hhae044.

Gu, S., Kelly, B. and Xiu, D., Empirical asset pricing via machine learning. *Rev. Financ. Stud.*, 2020, **33**(5), 2223–2273.
Hamilton, J. D. and Lin, G., Stock market volatility and the business cycle. *J. Appl. Economet.*, 1996, **11**(5), 573–593.
Hampel, F. R., The influence curve and its role in robust estimation. *J. Am. Stat. Assoc.*, 1974, **69**(346), 383–393.
Henkel, S. J., Martin, J. S. and Nardari, F., Time-varying short-horizon predictability. *J. Financ. Econ.*, 2011, **99**(3), 560–580.
Hoerl, A. E. and Kennard, R. W., Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics.*, 1970, **12**(1), 55–67.
Holopainen, M. and Sarlin, P., Toward robust early-warning models: A horse race, ensembles and model uncertainty. *Quant. Financ.*, 2017, **17**(12), 1933–1963.
Hong, H. and Stein, J. C., A unified theory of underreaction, momentum trading, and overreaction in asset markets. *J. Financ.*, 1999, **54**(6), 2143–2184.
Huang, D., Jiang, F., Tu, J. and Zhou, G., Investor sentiment aligned: A powerful predictor of stock returns. *Review of Financial Studies*, 2015, **28**(3), 791–837.
Inoue, A. and Kilian, L., In-sample or out-of-sample tests of predictability: Which one should we use? *Econom. Rev.*, 2005, **23**(4), 371–402.
Israel, R., Kelly, B. T. and Moskowitz, T. J., Can machines 'learn' finance? *J. Invest. Manag.*, 2020, **18**(2), 23–36.
Kelly, B., Malamud, S. and Zhou, K., The virtue of complexity in return prediction. *J. Financ.*, 2024, **79**(1), 459–503.
Kelly, B. and Pruitt, S., Market expectations in the cross-section of present values. *J. Financ.*, 2013, **68**(5), 1721–1756.
Kelly, B. and Xiu, D., Financial machine learning. *Found. Trends® Financ.*, 2023, **13**(3–4), 205–363.
Kolmogorov, A. N., On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Dokl. Akad. Nauk SSSR*, 1957, **114**(5), 953–956.
Kozak, S., Nagel, S. and Santosh, S., Shrinking the cross-section. *J. Financ. Econ.*, 2020, **135**(2), 271–292.
Leippold, M., Wang, Q. and Zhou, W., Machine learning in the Chinese stock market. *J. Financ. Econ.*, 2022, **145**(2), 64–82.
Leitch, G. and Tanner, J. E., Economic forecast evaluation: Profits versus the conventional error measures. *Am. Econo. Rev.*, 1991, **81**(3), 580–590.
Maio, P., Cross-sectional return dispersion and the equity premium. *J. Financ. Mark.*, 2016, **29**, 87–109.
Martin, I., What is the expected return on the market? *Quart. J. Econo.*, 2017, **132**(1), 367–433.
McCracken, M. W. and Ng, S., FRED-MD: A monthly database for macroeconomic research. *J. Bus. Econ. Stat.*, 2016, **34**(4), 574–589.
Neely, C. J., Rapach, D. E., Tu, J. and Zhou, G., Forecasting the equity risk premium: The role of technical indicators. *Manage. Sci.*, 2014, **60**(7), 1772–1791.
Olson, L. M., Qi, M., Zhang, X. and Zhao, X., Machine learning loss given default for corporate debt. *J. Empir. Financ.*, 2021, **64**, 144–159.
Pesaran, M. H. and Timmermann, A., A simple nonparametric test of predictive performance. *J. Bus. Econ. Stat.*, 1992, **10**(4), 461–465.
Pettenuzzo, D. and Timmermann, A., Predictability of stock returns and asset allocation under structural breaks. *J. Econom.*, 2011, **164**(1), 60–78.
Rapach, D. E., Ringgenberg, M. C. and Zhou, G., Short interest and aggregate stock returns. *J. Financ. Econ.*, 2016, **121**(1), 46–65.
Rapach, D. E., Strauss, J. K. and Zhou, G., Out-of-sample equity premium prediction: Combination forecasts and links to the real economy. *Rev. Financ. Stud.*, 2010, **23**(2), 821–862.
Rapach, D. and Zhou, G., Forecasting stock returns. In *Handbook of Economic Forecasting*, edited by G. Elliott and A. Timmermann, pp. 328–383, 2013 (Elsevier North Holland: Amsterdam).
Rossi, B. and Inoue, A., Out-of-sample forecast tests robust to the choice of window size. *J. Bus. Econ. Stat.*, 2012, **30**(3), 432–453.
Schwert, G. W., Why does stock market volatility change over time? *J. Financ.*, 1989, **44**(5), 1115–1153.
Spiegel, M., Forecasting the equity premium: Where we stand today. *Rev. Financ. Stud.*, 2008, **21**(4), 1453–1454.
Tibshirani, R., Regression shrinkage and selection via the Lasso. *J. R. Stat. Soc. B*, 1996, **58**(1), 267–288.
Timmermann, A., Forecast combinations. In *Handbook of Economic Forecasting*, pp. 135–196, 2006 (Elsevier).
Timmermann, A., Forecasting methods in finance. *Annual Review of Financial Economics*, 2018, **10**(1), 449–479.
Welch, I. and Goyal, A., A comprehensive look at the empirical performance of equity premium prediction. *Rev. Financ. Stud.*, 2008, **21**(4), 1455–1508.
Wolff, D. and Neugebauer, U., Tree-based machine learning approaches for equity market predictions. *J. Asset Manag.*, 2019, **20**(4), 273–288.
Zou, H. and Hastie, T., Regularization and variable selection via the elastic net. *J. R. Stat. Soc. B*, 2005, **67**(2), 301–320.

Appendix

A. Eleven Machine Learning Methods Used in Gu *et al.* (2020)

We first follow Gu *et al.* (2020) and divide the eleven machine learning methods into four groups based on their modeling techniques: dimension reduction, penalized regression, tree-based models, and neural networks.

(1) Dimension reduction: PCR and PLS
PCR and PLS are two classic dimension-reduction-based regressions. They both extract effective principal components from high-dimensional variables and then use them as explanatory variables to implement linear regression. The idea of PCR is to regularize the prediction problem by zeroing out the coefficient on low-variance components. The difference between PCR and PLS is that the PLS incorporates the target variable (that is the stock return r_{t+1}) in the dimension reduction step. Thus, PLS generates its principal components by conditioning on the joint distribution of the target variable and the predictors. To determine the number of principal components k for PCR and PLS, we follow Gu *et al.* (2020) and use a validation set to select the best number of principal components.

(2) Penalized regressions: LASSO and ENet
LASSO (Tibshirani 1996) and ENet (Zou and Hastie 2005) are two popular shrinkage methods for linear regression. They both aim to minimize the loss function of ordinary least squares with a penalization term. The difference between LASSO and ENet is that LASSO implements shrinkage with the L_1-penalized function while ENet uses both the L_1-penalized and L_2-penalized functions. Due to the introduction of L_1-penalized function, both models can perform variable selection and avoid overfitting. We use the validation set to select the best shrinkage parameter. It is noted that Gu *et al.* (2020) employ the Group LASSO method, which can be seen as a variation of LASSO. Recent studies show their strong predictive power in forecasting stock returns (Chinco *et al.* 2018, Feng *et al.* 2020, Freyberger *et al.* 2020, Kozak *et al.* 2020).

(3) Tree-based models: RF and GBRT
RF (Breiman 2001) and GBRT (Friedman 2001) are two commonly used regression tree models in machine learning literature. Constructed by many different trees, RF adopts a bagging ensemble method to reduce prediction variance. Thus, RF can always outperform a single decision tree model. Similarly, the GBRT employs a boosted strategy, which recursively merges many simple tree models. The GBRT can be interpreted as an optimization algorithm on an appropriate loss function. It combines many weak tree models into one strong complex tree model.

(4) Neural Networks: NN1 ∼ NN5
Among various machine learning methods, neural networks (NNs) have become increasingly popular and broadly used across different domains since NNs can theoretically approximate any continuous function (Kolmogorov 1957, Diaconis and Shahshahani 1984,

Cybenko 1989) and exhibit excellent predictive power. In this study, we focus on the typical multilayer perceptrons, which is a frequently used NN technique. Following Gu *et al.* (2020), we consider NN architectures with hidden layer depths ranging from 1 to 5 (NN1 ∼ NN5). NN1 comprises a single hidden layer containing 32 neurons. NN2 consists of two hidden layers with 32 and 16 neurons, respectively. NN3 features three hidden layers with 32, 16, and 8 neurons. NN4 includes four hidden layers with 32, 16, 8, and 4 neurons. Lastly, NN5 is structured with five hidden layers comprising 32, 16, 8, 4, and 2 neurons. We employ the rectified linear unit (ReLU) as the activation function and use the stochastic gradient descent algorithm to train the various NN models.

B. Hyperparameter Tuning

Table B1 summarizes the hyperparameter tuning for each machine learning model used in this study.

Table B1. Hyperparameter settings for fifteen machine learning models.

Forecasting models	Tuning parameters
PLS	The number of components: $\{1, 2, 3, 4, 5, 6, 7, 8\}$
PCR	The number of components: $\{1, 2, 3, 4, 5, 6, 7, 8\}$
LASSO	L1 regularization parameter: $[10^{-4}, 10]$
ENet	Constant that multiplies the penalty terms: $[10^{-4}, 10]$ The ENet mixing parameter: $\{0.2, 0.5, 0.8\}$
GBRT	The number of boosting stages to perform: $\{10, 50, 100, 150, 200\}$ Maximum depth of the individual regression estimators: $\{2, 3, 4\}$ The minimum number of samples required to be at a leaf node: $\{1, 3, 5\}$
RF	The number of trees in the forest: $\{10, 50, 100, 150, 200\}$ Maximum depth of the individual regression estimators: $\{2, 3, 4\}$ The minimum number of samples required to be at a leaf node: $\{1, 3, 5\}$
NN1 ∼ NN5	Dropout rate: $\{0.2, 0.4, 0.6, 0.8\}$ Learning rate: $\{0.001, 0.01\}$ L2 regularization parameter (also called weight decay): $\{0.1, 0.01, 0.001\}$
Ridge	L2 regularization parameter: $\{10^k \| k = 0, 1, 2, \cdots, 20\}$
SVR	The kernel type to be used in SVR: $\{\, '\text{linear}', '\text{poly}', '\text{rbf}', '\text{sigmoid}'\}$ Degree of the polynomial kernel function: $\{2, 3, 4\}$ L2 regularization parameter: $\{0.1, 0.5, 1\}$
KNR	Number of neighbors: $\{3, 4, 5, 6, 7\}$ Weight function used in prediction: $\{\, '\text{distance}', '\text{uniform}'\}$ Leaf size: $\{20, 30, 40\}$ Power parameter for the Minkowski metric: $\{1, 2, 3\}$
XGBoost	Maximum depth of a tree: $\{4, 5, 6, 7, 8\}$ Step size shrinkage used in update to prevents overfitting: $\{0.01, 0.1\}$ L2 regularization parameter: $\{0, 0.5, 1\}$ L1 regularization parameter: $\{0, 0.5, 1\}$

Notes: We use the default settings for other parameters in each machine learning method. For further details, refer to the corresponding module documentation in Python: the scikit-learn module (https://scikit-learn.org/0.21/documentation.html), the PyTorch module (https://pytorch.org/docs/stable/index.html), the xgboost module (https://xgboost.readthedocs.io/en/stable/index.html).