Man versus Machine Learning Revisited*

Yingguang Zhang Yandi Zhu Juhani T. Linnainmaa

May 2024

Abstract

Binsbergen et al. (2023) predict analysts' forecast errors using a random forest model. A strategy that trades against this model's predictions earns a monthly alpha of 1.54% (t-value = 5.84). This estimate represents a large improvement over studies that have used classic statistical methods. We attribute the difference to a look-ahead bias. Removing the bias erases the alpha. We show how feature importances and a diagnostic test can help detect whether machine-learning techniques might have a look-ahead bias. Consistent with Binsbergen et al. (2023), we find that machine learning-based forecasts are incrementally informative about future earnings.

*Yingguang (Conson) Zhang is with Peking University Guanghua School of Management (E-mail: yingguang.zhang@gsm.pku.edu.cn). Yandi Zhu is with Peking University School of Economics (E-mail: yandi.zhu@stu.pku.edu.cn). Juhani T. Linnainmaa is with Dartmouth College, NBER, and Kepos Capital (E-mail: juhani.t.linnainmaa@tuck.dartmouth.edu).

1 Introduction

Are sell-side analysts' forecasts biased and, if so, can investors profit by trading against such biases? La Porta (1996) and So (2013) find that strategies that bet against analysts' long-term growth expectations and estimated biases are profitable. Neither strategy, however, has earned a reliable alpha since 2000.[1] These results could be interpreted as implying that analysts' biases might have correlated with (or even contributed to[2]) mispricings but that markets have since turned more efficient. Indeed, after decades of research into analysts, the consensus has emerged that the link between biases in analyst expectations and stock prices is economically and statistically weak.[3]

Binsbergen et al. (2023) seemingly upend this literature. This study trains a random-forest regression to serve as "a statistically optimal unbiased machine-learning benchmark." By taking the difference between analysts' forecasts and this benchmark they measure how overly optimistic or pessimistic analysts are in each of their forecasts. Their approach appears to prove extraordinarily fruitful. A strategy that buys (sells) stocks that receive overly pessimistic (optimistic) earnings forecasts earns a monthly value-weighted Fama-French five-factor model alpha of 1.54% (t-value = 5.84) from 1986 to 2019.[4]

This finding stands out for three reasons. First, other studies that construct similar strategies find weaker results that grow weaker still in the more recent data.[5] Second, even

[1] Chen and Zimmermann's (2022) Open Source Asset Pricing replication project includes La Porta's (1996) predictor. In the post-2000 data, a value-weighted version of this strategy earns an average return of 17 basis points per month (t-value = 0.38). Binsbergen et al. (2023, Internet Appendix A.10) find that while the So (2013) strategy earns an average return of 111 basis points (t-value = 3.79) in the pre-2000 data, this average is 33 basis points (t-value = 1.79) in the post-2000 data.

[2] See, for example, Engelberg et al. (2020).

[3] Kothari, So, and Verdi (2016) review the analyst literature and conclude that "evidence on the relation between analysts' forecasts and expected returns is still scarce."

[4] This study has made an impact. As of April 2024, it has been cited over 80 times according to Google Scholar. Several published papers apply Binsbergen et al.'s (2023) model and/or alternative machine learning models to construct earnings expectations (e.g., Cassella et al., 2022 and Silva and Thesmar, 2024), and return expectations (Cao et al., 2024b). A number of working papers apply machine-learning models to study related topics (e.g., Campbell et al., 2024 and Cao et al., 2024a). Zhang et al. (2023) use Binsbergen et al.'s (2023) benchmark to measure the myopicity of analysts' expectations.

[5] Binsbergen et al. (2023), for example, contrast their and So's (2013) techniques and note that, unlike theirs, So's (2013) return effect is absent in more recent data: "We demonstrate that the variable selection

Table 1: Monthly Fama-French five-factor model alphas and factor loadings of portfolios sorted on analysts' conditional biases

			Replication			
	BHL Original		With Look-ahead Bias		Without Look-ahead Bias	
	Coef.	t-value	Coef.	t-value	Coef.	t-value
Intercept	−1.54	−5.84	−1.67	−5.95	−0.44	−1.63
MKT	0.38	5.28	0.43	5.81	0.43	5.86
SMB	0.61	5.17	0.66	5.17	0.76	5.85
HML	0.95	7.12	0.89	6.16	0.80	5.60
RMW	−0.68	−4.10	−0.75	−4.09	−0.70	−3.76
CMA	−0.53	−1.93	−0.30	−1.11	−0.15	−0.56

This table reports Fama-French five-factor model alphas and factor loadings for three long-short strategies. Each strategy is long stocks with the highest conditional earnings forecast biases and short stocks with the lowest biases. Columns 1 and 2 report the estimates from Table 6 in Binsbergen et al. (2023). Columns 3 and 4 show our replication of the result. Columns 5 and 6 remove the variable with the look-ahead bias from the random forest regression used to estimate the conditional biases. The sample period is January 1986 through December 2019.

outside the analyst literature their results dwarf *all* other anomalies in their sample. The Binsbergen et al. (2023) effect is an outlier by a significant margin among 148 anomaly portfolios over the 1986–2019 period. Third, Binsbergen et al. (2023) train the random forest regression using just one or two years of data at a time. If this technique outperforms So's (2013) protocol, it must be because the random forest regression uncovers meaningful interactions and nonlinearities from the data. However, with 75 features and a small sliver of training data, it seems, a priori, challenging for a random forest model to do so.[6]

If the Binsbergen et al. (2023) result is correct, it has far-reaching ramifications for the academic research process. As a first-order effect, any study that seeks to understand

step is not innocuous, and most (if not all) of the return predictability examined in So (2013) using linear forecasts disappears after the 2000s."

[6]The machine-learning literature typically calls dependent and independent variables "targets" and "features." We use this nomenclature when referring to the random forest regression. The trade-off between the amount of training data and the quality of the model's predictions depends on model complexity. In machine learning, a model's *capacity* measures its "ability to fit a wide variety of functions" (Goodfellow et al., 2016). A model with high capacity (such as a random forest) requires more data to avoid overfitting to the training data and to keep the potential benefits of high capacity when making predictions in the test data.

analysts' expectations (and the biases therein) must use or control for the Binsbergen et al. (2023) technology. Moreover, in the spirit of Harvey et al. (2016), a t-value of 5.84 sets a high bar for what constitutes a statistically and economically meaningful effect in the analyst literature. In short, the Binsbergen et al. (2023) result would affect what would be publishable and, therefore, how researchers might choose to allocate their efforts.

In this study we show that these inferences are premature. Table 1 explains our key point. In columns 1 and 2 we report the estimates from Binsbergen et al.'s (2023) Table 6. These estimates represent the profitability of a strategy that trades against predicted analyst forecast errors. (We provide the details of this strategy later.) This is the strategy that earns an alpha of -1.54% per month.[7] In columns 3 and 4 we replicate their result based on the details provided in the original paper and a search over plausible rules when the details are missing. We closely match the estimates from Binsbergen et al. (2023), not only in terms of the alpha estimates (-1.67% versus -1.54%) but also the factor loadings.

Alas, our replication (and, by extension, the results in Binsbergen et al. (2023)) has a look-ahead bias. One of the 75 features is the company's actual earnings, described in BHL (p. 14) as "Realized earnings from the last period." However, this variable is from the "last period" from the perspective of the earnings being forecasted. This is a problem: when the target variable is year-2 earnings, this variable is the firm's year-1 earnings, a variable unknown to econometricians and investors at time t.

This look-ahead bias is important. In columns 5 and 6 we remove this bias by using realized earnings from the most recent period as of the forecasting date. We keep every other step of the analysis unchanged. This modification reduces the monthly alpha to -44 basis points and we can no longer reject the null hypothesis that this strategy's true alpha

[7] This strategy's three-factor model alpha is -1.96% per month (t-value $= -8.64$). The three-factor model alpha exceeds the five-factor model alpha in magnitude because the strategy covaries negatively with the profitability (Bouchaud et al., 2019) and investment factors (Cooper et al., 2008); the factor loadings on RMW and CMA are -0.68 and -0.53. That is, analysts are overly optimistic (based on Binsbergen et al.'s (2023) bias measure) about unprofitable firms that invest aggressively. The five-factor model alpha measures how profitable it is to trade against predicted analyst biases net of the profitability and investment effects.

is zero (t-value $= -1.63$). The similarity between the factor loadings in columns 3 and 5 indicates that the look-ahead bias creates pure alpha unrelated to the risk factors. The remaining result, the slightly negative but statistically insignificant alpha, is consistent with studies such as La Porta (1996) and So (2013). It has historically been profitable to trade against analysts' forecasts, but far less so in the recent data.

In this study we both replicate and extend Binsbergen et al. (2023) and propose methods for detecting a look-ahead bias. Our key conclusions are as follows. First, the use of different implementations of the random forest model and alternative machine-learning techniques (such as gradient boosting) does not meaningfully alter the conclusion that the association between the conditional bias in analysts' expectations and future stock returns is weak. Second, the machine learning apparatus does not contribute to the alphas in BHL. Retaining the look-ahead bias but replacing the model with a parsimonious OLS model delivers, if anything, a higher alpha. Third, machine-learning techniques can improve the accuracy of earnings forecasts, although the removal of the look-ahead bias reduces this approach's efficacy. Fourth, we find that measures of feature importance and a diagnostic test can be useful for detecting the possibility that the machine learning model has a look-ahead bias.

We proceed as follows. In Section 2 we describe the replication process, compare our replication to Binsbergen et al. (2023), and show which parts of BHL are (and are not) affected by the look-ahead bias. In Section 3, we propose a diagnostic test for evaluating whether a machine learning model has access to information not available to econometricians in real time. We also show how feature importances coupled with parsimonious linear models can be used to measure the value add of "black box" machine-learning techniques. In Section 4 we employ alternative machine-learning techniques to try to rejuvenate the Binsbergen et al. (2023) result. Section 5 concludes.

2 Replication of Binsbergen et al. (2023)

2.1 Data and sample

We use data from the same four sources as Binsbergen et al. (2023) (BHL):

1. Monthly stock-level data from CRSP,
2. Analysts' consensus earnings forecasts and actual earnings from IBES Unadjusted Files,
3. Firm-level fundamental data from Financial Ratios Suite by Wharton Research Data Services,
4. Real-time macroeconomic indicators from the Federal Reserve Bank of Philadelphia.

BHL also graciously provided the conditional bias estimates from the original study. Our main sample covers U.S. common stocks (CRSP share codes 10 and 11) listed on the NYSE, AMEX, and NASDAQ (exchange codes 1, 2, or 3), running from January 1984 through December 2019. Following BHL, we examine analysts' forecasts of annual earnings at the one-year and two-year horizons (FPI = 1 and 2), and quarterly earnings forecasts at the one-quarter, two-quarter, and three-quarter horizons (FPI = 6, 7, and 8). We detail the sample construction procedure in Appendix B.

We follow BHL and use three sets of variables to predict firms' future earnings: firm characteristics, macroeconomic variables, and analysts' forecasts. As discussed in the introduction, the variable with the look-ahead bias is the first item on the list, "realized earnings in the last period." We list the individual variables in Appendix A.1.

1. **Firm characteristics**:
   (a) Realized earnings in the last period.
   (b) Monthly stock prices and returns from CRSP.
   (c) 67 financial ratios from the Financial Ratios Suite by Wharton Research Data Services. We fill missing values with industry medians based on the Fama-French

49 industry classification.[8]

2. **Macroeconomic variables**:[9]

   (a) Consumption growth, defined as the log difference of consumption in goods and services (RCON).

   (b) GDP growth, defined as the log difference of real GDP (ROUTPUT).

   (c) Growth of industrial production, defined as the log difference of Industrial Production Index (IPT).

   (d) Unemployment rate (RUC).

3. **Analyst forecasts**:

   (a) Analysts' consensus quarterly and annual earnings-per-share (EPS) forecasts.

This list has 75 characteristics that BHL use to forecast earnings. Because BHL model the term structure of expectations, they estimate models with different target variables: one-quarter-ahead (Q1), two-quarters-ahead (Q2), three-quarters-ahead (Q3), one-year-ahead (A1), and two-years-ahead (A2) earnings. BHL define realized earnings and analyst forecasts on the list above to match each regression's target variable. For example, when they forecast quarter-1 earnings, "realized earnings" is the prior quarter's earnings and analysts' forecast is about quarter-1 earnings. On the other hand, when they forecast year-2 earnings, "realized earnings" is the year-1 earnings and the forecast is about year-2 earnings. This treatment means that three of the five models—those with Q2, Q3, and A2 earnings as the target—have the look-ahead bias.

[8]BHL note in Online Appendix A.2 that they "consider another twenty-six fundamental values per share derived from these financial ratios." They do not enumerate these additional per-share characteristics.

[9]Federal Reserve Bank of Philadelphia issues monthly releases for consumption, real GDP, and industrial production, and quarterly releases for unemployment. We employ the latest vintages available each month.

2.2 Model specification and implementation details

We train a random forest model each month t to predict firms' EPS for each forecasting horizon h:

$$E_t[AE^h_{i,t}] = f^h_t\left(\boldsymbol{X}_{i,t}|\boldsymbol{\gamma}\right), \tag{1}$$

where $AE^h_{i,t}$ denotes stock i's actual EPS at horizon h relative to time t, and f^h_t denotes the random forest (RF) model that takes the feature set $\boldsymbol{X}_{i,t}$ as input with hyperparameters $\boldsymbol{\gamma}$. The forecasting horizon h takes the values of one quarter (Q1), two quarters (Q2), three quarters (Q3), one year (A1), and two years (A2).

We follow BHL's implementation. When the necessary details are missing, we do a grid search over all plausible choices and select the ones that yield results that better match those in BHL. When the results are very similar between different choices, we choose the more standard or simpler choice.[10] We describe our specification details below and summarize them in Table 2.

1. **Standardization**. BHL state in footnote 15 that they standardize the features. However, they do not provide a guideline for the standardization procedure. We try cross-sectional z-scores and ranks, as well as not standardizing. The replication results are similar under all these choices, with unstandardized features yielding slightly better-matching results.[11]
2. **Winsorization**. BHL state in footnote 15 that they winsorize the features at the 1% level. The code available on the journal's website also winsorizes the target variables. We experiment with only winsorizing the features and winsorizing both the features

[10] We present the results for alternative specifications in Section 4.1.

[11] Unlike regressions with regularization penalties, such as LASSO (L1 norm) and ridge (L2 norm) regressions, random forest regressions do not present an inherent need for feature normalization. However, cross-sectional normalization can make a difference in panel data in random forest regressions by eliminating shifts in the features' distributions.

and targets. The latter yields more closely matching results.

3. **Per-share data**. BHL state in Online Appendix A2 that they also consider 26 additional fundamental per-share characteristics, such as the book equity per share and the current debt per share to improve the forecasts. They do not, however, provide a full list of these per-share variables. We try both including and not including a set of per-share characteristics we deem plausible (see Table A1). The results are similar with and without these characteristics and we therefore exclude these variables from the main specification.

4. **Hyperparameters of the random forest regression**. A random forest regression is an ensemble of decision trees. Various hyperparameters, such as maximum depth and sample fraction, control what type of variation is induced in the data in the generation of the individual trees. The following choices are about these hyperparameters:

   (a) **Sample fraction**. BHL state in Table 1 that they use a sample fraction of 1% for training each decision tree. The code on the journal website uses a sample fraction of 5%. Using the 5% fraction produces a better match with the reported results.

   (b) **Feature fraction**. This parameter determines the number of features to be considered in each split. BHL state in Appendix A.1 that they always include the five most important features for every split to choose from. In the code posted on the journal website, they appear to include also 50% of all available features at random. However, this line is commented out, in which case the code would default to drawing the square root of the total number of features.[12] We consider two implementations. First, we use R and estimate the random forest regression

[12]The popular machine-learning packages in neither Python nor R gives an out-of-the-box method for estimating a random forest regression in which every split includes the five most important features. To implement this scheme, the following steps need to be followed: (1) train an initial random forest model with feature fraction set to 1.0; (2) identify the top-5 features based on feature importance, and (3) re-estimate the random forest model with the option of always including the top-5 features. The RANGER package in R, which the authors use in their code to estimate the random forest model, gives an ALWAYS.SPLIT.VARIABLES option of including a fixed set of features into every split.

twice. In the first stage we include all features; in the second stage we include the top-5 features and 50% of all features at random. Second, we use Python and set the feature fraction to 1.0 so that all splits can consider the full set of features and, therefore, the top-5 features are always available for each split. The two implementations yield very similar results, likely because only a very small number of features (i.e., fewer than five) are important for predicting earnings; that is, the features outside the top five are rarely chosen for splits.[13]

(c) **Number of trees**. We set the random forest size to 2,000 trees.

(d) **Maximum depth**. Each tree is limited to a maximum depth of 7.

(e) **Minimum node size**. We set the minimum number of samples required for further splitting a node to 5.

5. **The timing of the features.** BHL state in Appendix A3 that they require the forecasting variables to be available prior to the time when analysts' forecasts become available. Hence, when constructing the random forest forecasts in month t, we use analysts' forecasts available in month t and all other variables available at the end of month $t-1$.

6. **Training window**. We train the model monthly using a 12-month rolling window when forecasting quarterly and one-year-ahead earnings. We use 24-month rolling windows to forecast two-year-ahead earnings. These choices are the same as those reported in BHL.

2.3 Man versus machine earnings forecasts

Table 3 shows that we are able to replicate BHL's sample and earnings forecasts closely. In Panel A we report values from Table 2 in BHL. Panels B and C present our replication results with and without the look-ahead bias. Each panel has five rows, with each row

[13] We measure and discuss feature importances in Section 3.2.

Table 2: Replication details

Panel A: Choices experimented		
	BHL	Replication
Timing of last-period realized earnings	Unspecified	**With look-ahead: $t+h-1$** Without look-ahead: $t-1$
Standardization	Unspecified	Z-scores Ranks **None**
Winsorization	Features (text) Features and targets (code)	Features **Features and targets**
Per-share data	Unspecified	Include **Exclude**
Sample fraction	1% (text) 5% (code)	1% **5%**
Feature fraction	Top five features (text) plus 50% at random (code)	Top five features plus 50% at random **All features**
Panel B: Other details		
Feature timing	Analysts' forecast at time t and all other features at $t-1$	
Number of trees	2,000	
Maximum depth	7	
Minimum node size	5	
Training window	24 months for year-2 forecasts and 12 months for other forecasts	

This table reports the specifications of the replication. Panel A reports which details are unspecified in BHL and the set of choices tried. The bolded option corresponds to the main specification, chosen on the basis of providing a better match with the results in BHL and/or simplicity. Panel B reports the other implementation details provided in the BHL.

corresponding to a different forecasting horizon. In BHL all earnings-related variables—actual and forecasted—are in the units of earnings-per-share (EPS).

[Table 3 about here]

Columns 1, 2, and 3 report the sample averages of random forest forecasts (RF), analysts' forecasts (AF), and actual EPS (AE). Columns 4 and 5 report the average differences between the RF forecasts and actual earnings and the t-values associated with these differ-

ences. The AF-minus-AE differences in columns 6 and 7 indicate that analysts' forecasts are systematically upward biased; the differences are positive and statistically significantly different from zero at all horizons. The random forest-based forecasts in column 4, by contrast, are close to zero.

We successfully replicate these estimates. All estimates in Panel B (with a look-ahead bias) are close to the BHL numbers in Panel A. Moreover, Panel C shows that the look-ahead bias does not affect this result; the random forest regression-based forecasts are unbiased also when we lag actual earnings in the models corresponding to horizons Q2, Q3, and A2 so that they would be known at the forecasting time.

Columns 8 and 9 report the mean-squared forecasting errors to assess how accurate the random forest model is relative to analysts. The result in BHL is that, by removing analysts' biases, the random forest model consistently exceeds the accuracy of the analysts. Our results in Panels B confirm this finding. However, in a small discrepancy to the BHL estimates, our estimated mean-squared errors are smaller for both analysts and the random forest model.[14]

Panel C shows that removing the look-ahead bias reduces the random forest model's accuracy at the affected horizons (Q2, Q3, and A2). Both in BHL and in our replication the random forest model with a look-ahead bias has the greatest advantage over analysts in the task of predicting year-2 earnings. Panel B shows that, at this horizon, the random forest model decreases mean-squared errors by 0.209 units—11.4%—from the analyst benchmark. However, Panel C shows that removing the look-ahead bias shrinks this gap to 0.049. That is, we attribute three-quarters of the random forest model's superior performance to the look-ahead bias. The degradations at the two other affected horizons (two- and three-quarters-ahead forecasts) are 37.5% and 60%. These estimates are consistent with the data leakage improving the model's forecasts. Nevertheless, consistent with BHL, the machine learning-based forecasts are more accurate than analysts at all five horizons even absent the

[14]Despite our best efforts, we could not identify the source of this discrepancy. In particular, by using 2020 vintages of the IBES and CRSP data, we confirmed that this discrepancy is not due to the changes made to the IBES and CRSP data since that date.

look-ahead bias.

2.4 Conditional bias and stock returns

Table 4 presents our replication of BHL's tests that predict the cross-section of stock returns. At each month t, we measure the biases in analysts' forecasts as the differences between the analyst (AF) and random forest (RF) forecasts, scaled by the month $t-1$ closing stock price:

$$Bias^h_{i,t} = \frac{AF^h_{i,t} - RF^h_{i,t}}{Price_{i,t-1}}, \tag{2}$$

where $AF^h_{i,t}$ and $RF^h_{i,t}$ are analysts' and random forest forecasts in month t for firm i's h-period-ahead EPS. We then average the biases across all horizons and use this "Average BE" to predict returns.[15] The trading strategy sorts stocks into quintiles based on the Average BE every month and buys (sells) stocks with high (low) estimated bias. This process is identical to that in BHL.

The first two rows of Table 4 show the results from Table 5 in BHL. Their results indicate that a monthly rebalanced value-weighted low-bias (quintile 1) portfolio earns an average return of 1.32% per month with a t-value of 6.53. The portfolio of the high-bias stocks (quintile 5) earns an average of -0.14% per month with a t-value of (-0.35). The difference between the bottom and top quintile portfolios' average returns is -1.46% with a t-value of -5.11. The third and fourth rows show the estimates from our replication with the look-ahead bias. Our replication produces a strategy that is equally profitable. The difference between the top and bottom quintiles is -1.55% with a t-value of -4.95.[16]

The BHL strategy is exceptionally profitable given that the sample runs from 1986

[15] We follow BHL and compute "Average BE" only for firm-month observations in which the bias estimate is non-missing for at least two of the five horizons.

[16] We follow BHL and use White's (1980) heteroskedasticity-robust standard errors to compute the test statistics.

through 2019 (many anomalies are weaker in post-2000 data) and leans towards larger stocks by the virtue of requiring analyst coverage for multiple horizons (many anomalies are weaker among larger firms).[17] Figure 1 shows the distribution of the annualized Sharpe ratios of 148 anomalies covered in Chen and Zimmermann's (2022) database. We make these anomalies comparable with the BHL strategy by using consistent samples and portfolio rules. The red vertical line indicates the Sharpe ratio of the BHL strategy, 0.95.[18] Compared to all other anomalies in the Chen and Zimmermann's (2022) database, BHL stands out; the second highest maximum annual Sharpe ratio in the sample is 0.76. The BHL strategy also clears with ease Ross's (1976) boundary for what constitutes a near-arbitrage opportunity; its squared Sharpe ratio ($0.95^2 = 0.90$) is more than twice that of the market ($0.43^2 = 0.18$).

Most of the profits of this trading strategy, however, stem from the look-ahead bias. Rows 5 and 6 show that when we compute average BEs without the look-ahead bias, the difference between the top and bottom quintiles decreases by four-fifths to −27 basis points. This average return is economically small and statistically indistinguishable from zero.

[Table 4 about here]

Figure 2 shows cumulative returns for three long-short strategies that trade against estimated biases. The blue line uses the estimates from BHL and the orange dashed line uses the estimates from our replication with a look-ahead bias. The two strategies move in lockstep and, over the full sample period, the correlation in monthly returns between the two is 0.94. The green dotted line, which is from the replication without the look-ahead bias, is unprofitable throughout the entire sample.

[Table 5 about here]

[17] See, for example, Green et al. (2017) and Hou et al. (2020).

[18] In this computation we use the data on conditional biases provided by BHL to compute the strategy's Sharpe ratio.

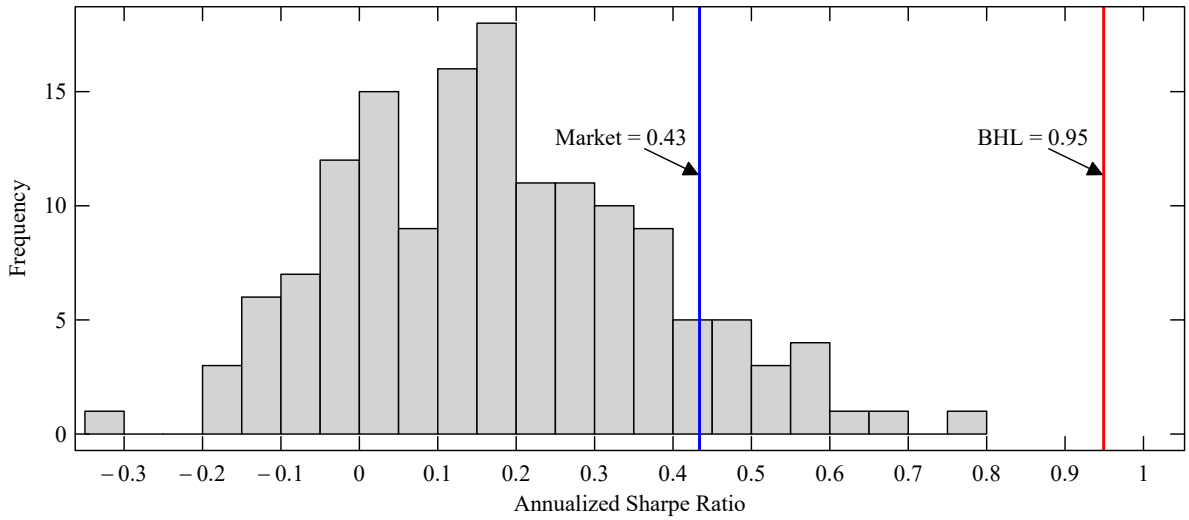


Figure 1: **Sharpe ratios of 148 anomaly portfolios**. This figure plots the distribution of annualized Sharpe ratios of 148 anomalies from Chen and Zimmermann's (2022) database. We impose the following restrictions both on the BHL strategy and all anomalies. The sample consists of common shares traded in NYSE, NASDAQ, or AMEX that have a positive book value of equity in the prior year and a share price above one dollar in the prior month. We also exclude microcap stocks, defined as those with a market value of equity below the 20$^{\text{th}}$ NYSE percentile in the prior month. We require the anomaly variables to be continuous (that is, excluding discrete and categorical variables) and available from 1986 to 2019 for at least 500 stocks in the average month. We augment the Chen-Zimmermann database with the price, size, and short-term reversal signals which it omits. All anomalies are value-weighted long-short portfolios based on monthly rebalanced quintiles. The red and blue vertical lines indicate the Sharpe ratios of the BHL strategy and the market portfolio. The Sharpe ratio of the market portfolio is computed from July 1926 to December 2020.

In Table 5 we measure the performances of the long-short strategies after controlling for common risk factors. Panels A, B, and C show BHL's original results, our replication results with and without the look-ahead bias. Panels A and B show that both the BHL strategy and our replication with the look-ahead bias earn large and statistically highly significant alphas under the CAPM and the Fama-French three- and five-factor models. Our replication closely matches BHL's estimates in terms of both alphas and factor loadings. Panel C shows that, after removing the look-ahead bias, the profits attenuate. While this strategy's CAPM and three-factor model alphas remain statistically significant, this remaining profitability is due to the correlations with the investment and profitability factors—analysts are more positive about unprofitable stocks that invest aggressively (Bouchaud et al., 2019; Cooper

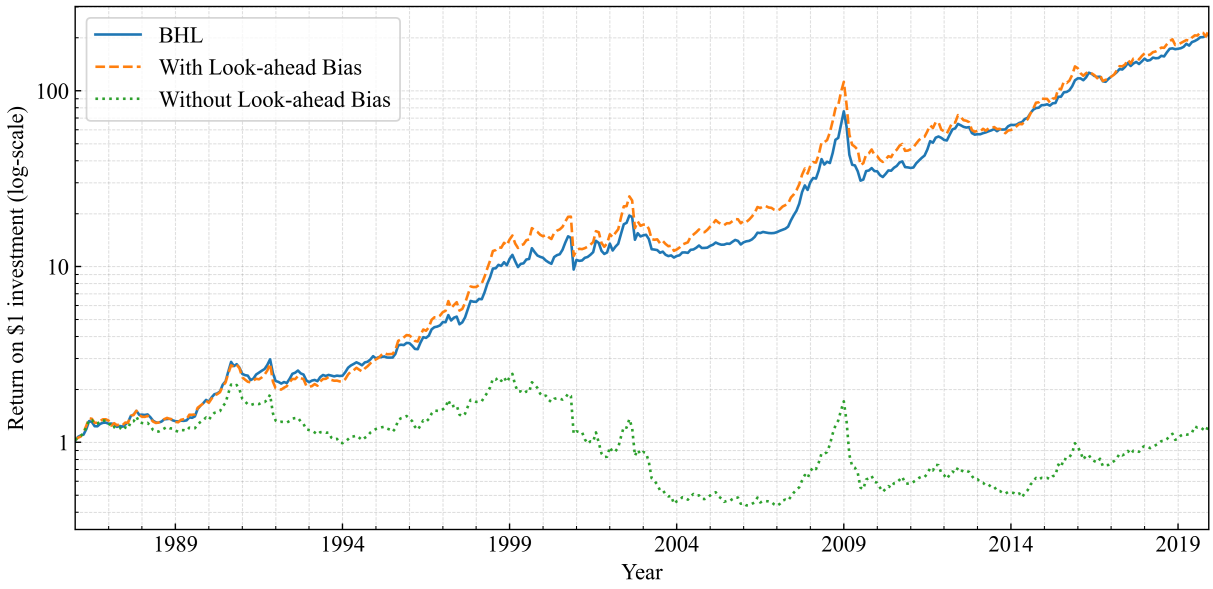


Figure 2: **Cumulative returns of value-weighted portfolios formed on conditional forecast biases.** This figure plots cumulative returns for value-weighted strategies that buy (sell) stocks in the lowest (highest) "Average BE" quintile and rebalance monthly. The three lines correspond to three implementations of this strategy: (a) The estimates using BHL's data (blue solid line); (b) our replication with a look-ahead bias (orange dashed line); and (c) our replication without a look-ahead bias (green dotted line). The sample period is January 1986 through December 2019.

et al., 2008). The five-factor model alpha of -0.44% per month (t-value $= -1.63$) shows that, net of these factor exposures, a strategy that trades against the estimated biases is not abnormally profitable.[19]

3 Detecting Look-ahead Bias in Complex Models

Look-ahead bias is a common pitfall in academic research.[20] A look-ahead bias means that, when predicting a variable such as asset return, a researcher uses future information not available at the time the prediction is made. The predictive power of a variable subject to

[19] We follow BHL in Tables 4 and 5 and analyze strategies that sort stocks based on *average* biases across the five horizons. In column RF in Table A.2 we report average returns and Fama-French five-factor model alphas for five separate strategies, one for each horizon. We find that none of these individual strategies earns a statistically significant average return or alpha.

[20] Bianchi et al. (2021), Cakici et al. (2024), Dickerson et al. (2023), and Duarte et al. (2023) are examples of recent papers that have identified studies in which look-ahead bias significantly affects the results.

the look-ahead bias can be large, yet often irrelevant from the investing perspective because investors can not act on this information.

The increases in the complexities of the statistical techniques employed and the amounts of data used may make it more difficult to diagnose look-ahead biases. For example, if an ensemble method such as random forest is treated as a black box and no attempt is made to understand how the features get transformed into predictions, it is nearly impossible to learn from the output alone whether a look-ahead bias might contribute to the results. This detection challenge is important because the potential for a look-ahead bias is likely greater when researchers work with novel datasets and methodologies; researchers in these situations may have less precise priors about what is and is not plausible. In this section we propose different ways of diagnosing complex models for the purposes of detecting the presence of a look-ahead bias.

3.1 Regression-based diagnosis

Suppose that $\hat{x}$ is trained to predict future x; $\hat{y}$ is trained to predict future y; and $\hat{x}$ and $\hat{y}$ are trained using similar but not identical data. The researcher is concerned that one of the signals may contain a look-ahead bias, that is, the data used to train $\hat{x}$ may contain future information about y or vice versa. We assume that the researcher does not have the training data but only the generated signals, $\hat{x}$ and $\hat{y}$, the realized values of x and y, and a small number of known ex ante determinants of x and y.

How can a researcher assess whether future information might have leaked into $\hat{x}$ or $\hat{y}$? The intuition of our diagnostic test is the following: $\hat{y}$ should not outperform $\hat{x}$ in predicting x after controlling for known ex ante determinants of x and y. This idea is best illustrated with an example. BHL construct forecasts for five horizons: Q1, Q2, Q3, A1, and A2, of which three (Q2, Q3, and A2) contain a look-ahead bias. Let us consider the Q1 and Q2 random-forest forecasts, RF^{Q1} and RF^{Q2} as $\hat{x}$ and $\hat{y}$. Because the first predictor (RF^{Q1})

is purposefully trained to predict Q1 earnings, it should predict these earnings better than RF^{Q2} which is trained for a *different* task using almost the same set of features. To assess whether RF^{Q2} contains one-quarter-ahead realized earnings (AE^{Q1}), we can estimate the following regression:

$$AE_{i,t}^{Q1} = \alpha + \beta_1 RF_{i,t}^{Q1} + \beta_2 RF_{i,t}^{Q2} + \underbrace{\gamma_1 AF_{i,t}^{Q1} + \gamma_2 AF_{i,t}^{Q2}}_{\text{Controls}} + \epsilon_{i,t}, \tag{3}$$

where $AE_{i,t}^{Q1}$ is the one-quarter-ahead realized earnings; $RF_{i,t}^{Q1}$ and $RF_{i,t}^{Q2}$ are the random forest forecasts for one-quarter- and two-quarter-ahead earnings; and $AF_{i,t}^{Q1}$ and $AF_{i,t}^{Q2}$ are analysts' forecasts as control variables. The regression coefficients β_1 and β_2 represent the marginal effects of the random-forest forecasts when controlling for analyst forecasts.

[Table 6 about here]

The first column in Table 6 reports the estimates from regression (3) using the random-forest forecasts computed *without* the look-ahead bias. The forecast from the random forest model which has been trained to predict Q1 earnings enters with a slope of 0.74 (t-value = 11.71); the forecast from the model trained to predict Q2 earnings has a slope of just 0.05 (t-value = 1.25). This result is unsurprising and consistent with the intuition above. Both predictors have access to much of the same information. Because RF^{Q1} is trained to this very task, it enters the regression with an economically large and statistically significant slope. The "mismatched" predictor RF^{Q2}, which is trained to a different task, is not incrementally informative about Q1 earnings.

The result in column 1 is specific to the situation without a look-ahead bias. In column 2 we use BHL's forecasts—labeled BHL^h and BHL^{h+1}—to estimate the same regression. A counterintuitive result emerges: the random forest forecast of Q2 earnings is a significant predictor of actual Q1 earnings (t-value = 13.22) while the "correct" forecast enters the regression with a negative sign (t-value = -2.09). What makes the Q2 forecast valuable

in this regression is the look-ahead bias, that is, the underlying model's access to future information, the actual Q1 earnings. Because BHL^{h+1} is a function of AE^{Q1}, it is not surprising that BHL^{h+1} is informative of AE^{Q1}. Column 3 shows that a regression in which we use our replication with a look-ahead bias produces similar estimates.

The other columns in Table 6 show the results for the other horizons in BHL that are subject to the look-ahead bias. When the dependent variable is the actual Q2 earnings, the random forest forecast of Q3 earnings is valuable *when* this variable has access to the actual Q2 earnings via the look-ahead bias; and when the dependent variable is the year-1 earnings, the year-2 is important for the same reason. We observe the same patterns when we use the BHL data and our replication with the look-ahead bias. The counterintuitive patterns disappear when we remove this bias.

Although this test is simple, we caution that, ex ante, it can be difficult to formulate if we do not already understand the nature of the potential look-ahead bias—which would, of course, render the test moot. In our application, for example, it is important to control for analysts' h and $h+1$ forecasts because when BHL train their models, they include AF^h only in the h model and AF^{h+1} only in the $h+1$ model. Because AF^{h+1} could be incrementally informative about period h earnings, a model trained with this information could add value. By controlling in the diagnostic regression for the two analyst forecasts, the random forest forecasts effectively have access to the same information.

The basic idea of the diagnostic test is to cross-check one signal against another signal. This test works when one of the signals does not have, or has less of, a look-ahead bias. In the first three regressions in Table 6, the divide is clear: the Q2 forecast has a look-ahead bias while the Q1 forecast does not. However, when the dependent variable is the actual Q2 earnings, *both* predictors have a look-ahead bias: the Q2 prediction has access to Q1 earnings while the Q3 prediction has access to Q2 earnings. However, in the task of predicting AE^{Q2}, the first of these look-ahead biases is far less important than the look-ahead bias in BHL^{Q3}.

Therefore, this test alone would (correctly) alarm the researcher about the issue in BHL^{Q3} but would (unfortunately) not raise concerns about BHL^{Q2}.

One can alternatively perform this test by regressing ex-post forecast error on predicted forecast errors:

$$FE_{i,t}^{h} = \alpha + \beta_1 BE_{i,t}^{h} + \beta_2 BE_{i,t}^{h+1} + \epsilon_{i,t}, \tag{4}$$

where $FE_{i,t}^{h} = AF_{i,t}^{h} - AE_{i,t}^{h}$ is analysts' ex post h-period-ahead forecast error and $BE_{i,t}^{h}$ and $BE_{i,t}^{h+1}$ are the conditional biases predicted by the models at horizons h and $h+1$. We can estimate regression (4) without the control variables because the differences in the two models' feature sets are unlikely to correlate with realized forecast errors. If $BE_{i,t}^{h+1}$ contains $AE_{i,t}^{h}$ in its construction, we would expect $BE_{i,t}^{h+1}$ to outperform $BE_{i,t}^{h}$ in predicting $FE_{i,t}^{h}$. We present the results from this alternative test in Appendix Table A.3 and find a pattern identical to that in Table 6.

These diagnostic tests can be applied to other settings in which the researcher is forecasting any variables x and y that are sufficiently related. For example, x and y can be firm investments or dividends at one- and two-year horizons or return and volatility at the same horizon. Although the models and data that are used to form predictions evolve, studies often make predictions about similar or cross-sectionally correlated variables; that is, asset pricing papers continue predicting the moments of returns and corporate finance studies remain interested in outcomes related to firm and management behavior. If a study proposes a new predictor $\hat{y}$, the spirit of our test is to add this predictor to the right-hand side of an old regression that tries to predict x with $\hat{x}$. If $\hat{y}$ is highly informative about x—a prediction task to which it has *not* been trained—the new predictor might warrant a closer look.

Although this diagnostic test is not perfectly revealing, it has two advantages. First, it is easily implementable. It does not require any information from "inside the black box" (such as a random forest model) that was used to generate $\hat{y}$. Second, the test is easily

interpretable. It relies on the simple logic that a new signal trained to task A should not be exceptionally well-suited for task B. A new predictor of future returns, for example, likely should not predict future earnings better than signals specifically trained to this task.

3.2 Diagnosis via feature importance

One of linear models' benefits over ensemble methods such as the random forest is their interpretability. Slopes represent explanatory variables' marginal effects and we can use standard statistical tests to evaluate these effects' statistical significance. Ensemble methods elude simple inteprerations. In a random forest a variable can, for example, influence the outcome positively or negatively depending on when a tree splits on it; and a variable can be unimportant in isolation but important through its interactions with other features. Researchers have proposed various measures of variable importance to overcome this limitation of ensemble methods, and it remains an active area of research.[21]

Feature importance can serve as a diagnostic tool. If a model has good predictive power, it is natural to ask which variables contribute to this power. There are two extreme cases of interest:

Case A: All features of the model are of approximately equal importance.

Case B: The model's predictive power stems from a handful of variables.

In the latter case we could, after the fact, replace the full model with a more parsimonious model with little loss in the model's fidelity. Feature importance can provide hints about possible data errors. A feature with a look-ahead bias can be of disproportional importance.

In Figure 3 we report feature importance for the five random forest models employed in BHL,[22] each corresponding to a different forecasting horizon. In this analysis we use the models trained with the look-ahead bias. We measure feature importance by estimating

[21] See, for example, Breiman (2001) and Verdineilli and Wasserman (2023).

[22] We follow BHL and normalize the feature importances to sum to one.

the model with the feature fraction set to 1.0 so that all splits consider all features.[23] The estimates show that the most important feature by far is the analyst forecast in four of the models. In the last model that predicts A2 earnings, lagged earnings (which has the look-ahead bias), is more important. Most of the other 73 features across all five models are of approximately zero importance.

The models corresponding to Q2, Q3, and A2 horizons in Figure 3 have the look-ahead bias. How much of the importance of the "Last EPS" variable is due to this bias? In Figure 4 we report estimates models without the look-ahead bias. This correction reduces the importance of "Last EPS." For example, in the regression that predicts A2 earnings, instead of being the most important feature, this feature now ranks distant third after analyst forecast and stock price ("prc").

The estimates in Figure 3 suggest that a parsimonious model with a handful of features would likely perform as well as the full model. We test this idea in Table 7. Instead of using the random forest models to make predictions each month, we only look up feature importances from these models. We then use the same training sample to estimate linear regressions with the top-five features and compute predictions from these parsimonious models.

[Table 7 about here]

A trading strategy based on the average bias implied by this linear model earns abnormal returns as large as the full random forest model. The abnormal returns of this strategy are highly significant across all factor models and both pre- and post-2000. The full-sample five-

[23] Figure 6 in BHL report feature importance one-quarter- and one-year-ahead horizons. Because they compute feature importance by looking at a model with a feature fraction set to 0.5 or $\sqrt{N_{\text{features}}}$, their estimates are more uniform than the ones in Figure 3. We replicate Figure 6 and extend it to include the two-quarter-, three-quarter-, and two-year-ahead earnings forecasts with and without look-ahead bias in Appendix Figure A.1. A value less than 1.0 for feature fraction inflates the importance of less meaningful features because each split considers only a random subset of all features. For example, if a tree can consider neither the analyst forecast nor realized earnings in the first split, it select some other variable that reduces the mean-squared error the most. After the fact the feature importance computation gives this variable credit for this reduction although the tree likely would have preferred to split on one of the dropped features.

factor model alpha of −1.81% (t-value = −6.34), for example, even *exceeds* the estimate in Panel B of Table 5, an alpha of −1.67% (t-value = −5.95). This result indicates, first, that the random forest model obtains most of its predictive power from just a handful of features and, second, that the very nature of the model—a regression tree can represent interactions and non-linearities—does not contribute to the model's predictive power. This computation illustrates how feature importance together with a parsimonious model can serve as another diagnostic tool. If a linear model with a few standard characteristics performs surprisingly well, then the results from the full ensemble model are likely not from the technology but from a data error.[24]

4 Alternative Machine Learning Forecasts

The two key results in BHL are about using random forest regressions to generate accurate and unbiased estimates of future earnings and a strategy that trades against predicted analyst biases. In terms of the first result, Table 3 shows that although the look-ahead bias enhances the random forest model's accuracy, these models remain more accurate than analysts even absent the look-ahead bias. In terms of the second result, Table 5 shows that almost all of the trading strategy's alpha derives from the look-ahead bias. These findings, however, leave open the possibility that perturbations to the specification of the

[24]BHL perform a similar robustness check using linear models with all features in Online Appendix Table A13. They report similar but weaker results. We trace this difference to their normalization procedure. In Table A.4 we examine the effects of feature selection and normalization on the estimates. (The normalization converts all explanatory variables into z-scores by subtracting off the cross-sectional mean and dividing by the cross-sectional standard deviation.) Panel A reports the estimates from Table 7 for reference. Panel B shows that using the top-5 features but with normalization degrades the strategy's efficacy. Its five-factor model alpha attenuates from −1.81% (t-value = −6.34) to −1.52% (t-value = −4.03). Panel C shows that using the full set of 75 features without normalization results in a strategy very similar to Panel A baseline with an alpha of −1.67% (t-value = −5.56). Panel D, which also normalizes the features and is therefore a replication of Online Appendix Table A13 in BHL, further degrades the alpha. The degradations from normalization are particularly pronounced in the post-2000 sample. With normalization, neither the top-5 features nor the full features strategy earns a statistically significant alpha in this period. We note that there is no inherent need for normalization in a linear model; a normalization is typically preferable in ridge, LASSO, and elastic net regressions because the regularization penalties treat all coefficients symmetrically. The downside of *cross-sectional* normalization can be that it destroys all time-series information.

random forest regression might resurrect the strategy's alpha, or that perhaps an alternative machine-learning technique might do so.

4.1 Alternative random forest specifications

We vary the random forest specification to examine whether its forecasting performance can be improved. We start with our baseline specification and vary the following options one at a time: sample fraction, feature fraction, the length of the training window, standardization, and adding per-share data.

[Table 8 about here]

Table 8 presents the results. Panel A shows the time-series average mean-squared earnings forecast errors at different horizons. Column 1 shows the baseline results from Table 3. In all columns starting from column 2, none of the forecast errors are noticeably smaller than the baseline numbers. A few of the specifications produce much larger forecast errors, such as when the sample fraction is set to 0.01 and the number of features is set to the floor of the square root of the total number of features, $\lfloor\sqrt{75}\rfloor = 8$.

Panel B shows the performance of the long-short portfolio formed on the average conditional biases. Across nearly all model specifications, the long-short portfolio that buys (sells) stocks with overly optimistic (pessimistic) forecasts earns negative abnormal returns. However, none of these portfolios perform meaningfully better than the baseline in column 1 or earn a significant five-factor model alpha. These results suggest that our main findings are robust to different random forest specifications.

In Appendix Table A.5 we present results from a more exhaustive search of alternative specifications. In this analysis we search over a grid of 56 alternative models; this grid has all permutations of the sample and feature fractions, the length of the estimation window, feature standardization, and the inclusion of the per-share features. We deem all of these 56

choices a priori equally plausible. The distributions of the alphas and their t-values do not provide evidence that any corner of the specification space displays statistically significant alphas. The median t-value across all specifications is -1.11 and the minimum is -2.13. (We are interested in the minimum because the BHL strategy buys high-bias stocks and sells low-bias stocks.) Under any standard correction for multiple hypothesis testing even the most extreme estimate does not support rejecting the null hypothesis of zero alpha at conventional levels of statistical significance.[25]

4.2 Alternative machine learning models

We next examine whether alternative machine learning models can deliver better performance. Following studies such as Gu et al. (2020) and Dong et al. (2022), we consider these alternative models:

- Ordinary least squares (OLS),
- Partial least squares (PLS, Wold, 1966),
- LASSO regression (LASSO, Tibshirani, 1996),
- Elastic net regression (ENet, Zou and Hastie, 2005),
- Random forest regression (RF, Breiman, 2001), and
- Gradient boosting decision tree (LightGBM, Ke et al., 2017).

We also consider "stacking," which we implement by averaging the forecasts across all models.

Each of these procedures except for the least squares requires us to specify hyperparameters. For example, in the case of elastic net regression, we need to specify the strengths of the L1 and L2 regularization penalties. We modify the investigation procedure followed in BHL and use time-series cross-validation to tune the hyperparameters each time we fit the

[25] A t-value of 2.13, absent a correction for MHT, associates with a p-value of 0.033. Both the Bonferroni and Holm-Bonferroni corrections would multiply this p-value by the total number of tests. Although the 56 tests in Table A.5 are correlated, as long as the "effective" number of tests is greater than three, none of the test statistics attain statistical significance at conventional levels.

models. We report further details in Appendix C.

The time-series cross-validation process is as follows. For each year t, we divide data up to year t into three partitions: the training sample (from year $t - w_{train} - w_{valid} - 1$ to $t - w_{valid} - 1$), the validation sample (from year $t - w_{valid} - 1$ to $t - 1$), and the test sample in year t. Variables w_{train} and w_{valid} denote the lengths of the training and validation windows. We fit the model on the training sample and then use the R^2 in the validation set to identify the optimal hyperparameters. With the selected hyperparameters, we retrain the model using both the training and validation samples to make predictions in the test sample. Because these machine learning algorithms are computationally intensive, we refit our model every year instead of every month.

We set the length of the training window to three years. Depending on the forecasting horizon, we adjust the validation window w_{valid} accordingly. We use a one-year window for the quarterly and one-year forecasts and the two-year window for the two-year forecasts. This modeling framework requires at least five years of data. The models therefore make the first predictions at the end of January 1989.

[Table 9 about here]

Panel A of Table 9 shows the mean-squared forecasting errors for different horizons. We find that the regularized regressions, such as LASSO and ENet, marginally improve the models' performance over ordinary least squares (OLS). The tree-based models (RF and LGBM), however, outperform linear models, especially in the task of forecasting longer-horizon earnings. For example, whereas the OLS model's average mean-squared error in forecasting two-year-ahead earnings is 1.840, both RF and LGBM reduce this error by approximately 10% to 1.640. Stacking different models produces the lowest forecasting error across all horizons. These results suggest that varying machine learning implementations can help improve the forecasting performance for earnings.

In Panel B of Table 9 we report on the performance of the long-short strategies that

trade against the biases predicted by different models. Similar to the results in Table 8, we do not find any of these alternative long-short portfolios perform noticeably better than the baseline specification. None of the five-factor model alphas is statistically significant at 5% level.

5 Conclusions

We revisit Binsbergen et al.'s (2023) findings on the efficacy of machine learning models in unbiasing analyst forecasts and extracting conditional biases from these forecasts. We confirm that machine-learning techniques can produce unbiased earnings forecasts and improve analysts' forecast accuracy, though this improvement is limited. However, the conditional biases implied by these models do not meaningfully predict abnormal stock returns in the cross-section after removing a look-ahead bias in Binsbergen et al.'s (2023) implementation.

Empirical research faces the rising challenge of replicating signals generated by complex models. We propose a simple diagnostic test that can detect look-ahead biases by cross-checking signals against each other. The intuition for our test is about task specificity. A predictor trained to task A should not perform as well in task B as predictors trained to that task. This test successfully detects the look-ahead bias in BHL using the model outputs alone.

A complication with ensemble methods, such as random forests and gradient boosting, is their lack of interpretability compared to traditional statistical methods. We show that feature importance coupled with parsimonious linear models can help detect data issues such as look-ahead biases.

Our findings have implications for future research. BHL's signal may lack predictive power for two reasons. First, analysts' biases may not incrementally explain the cross-section of stock returns. If so, our findings are consistent with the view of a semi-strongly efficient

financial market where sophisticated statistical techniques applied to public information may help improve investment performance at the margin, but near-arbitrage opportunities are ruled out. Under this view, the lack of predictability is not specific to the method or data used to extract the bias. Second, analysts' predictable biases may be important but machine learning models do not easily generate unique insights into this bias structure that the extant literature has missed. To this end, our findings suggest the deepening of economic understanding remains very much needed in the age of machines for knowledge creation.[26]

[26]This proposition aligns with recent work such as Cassella et al. (2022), Cao et al. (2024b), Silva and Thesmar (2024), and Zhang et al. (2023).

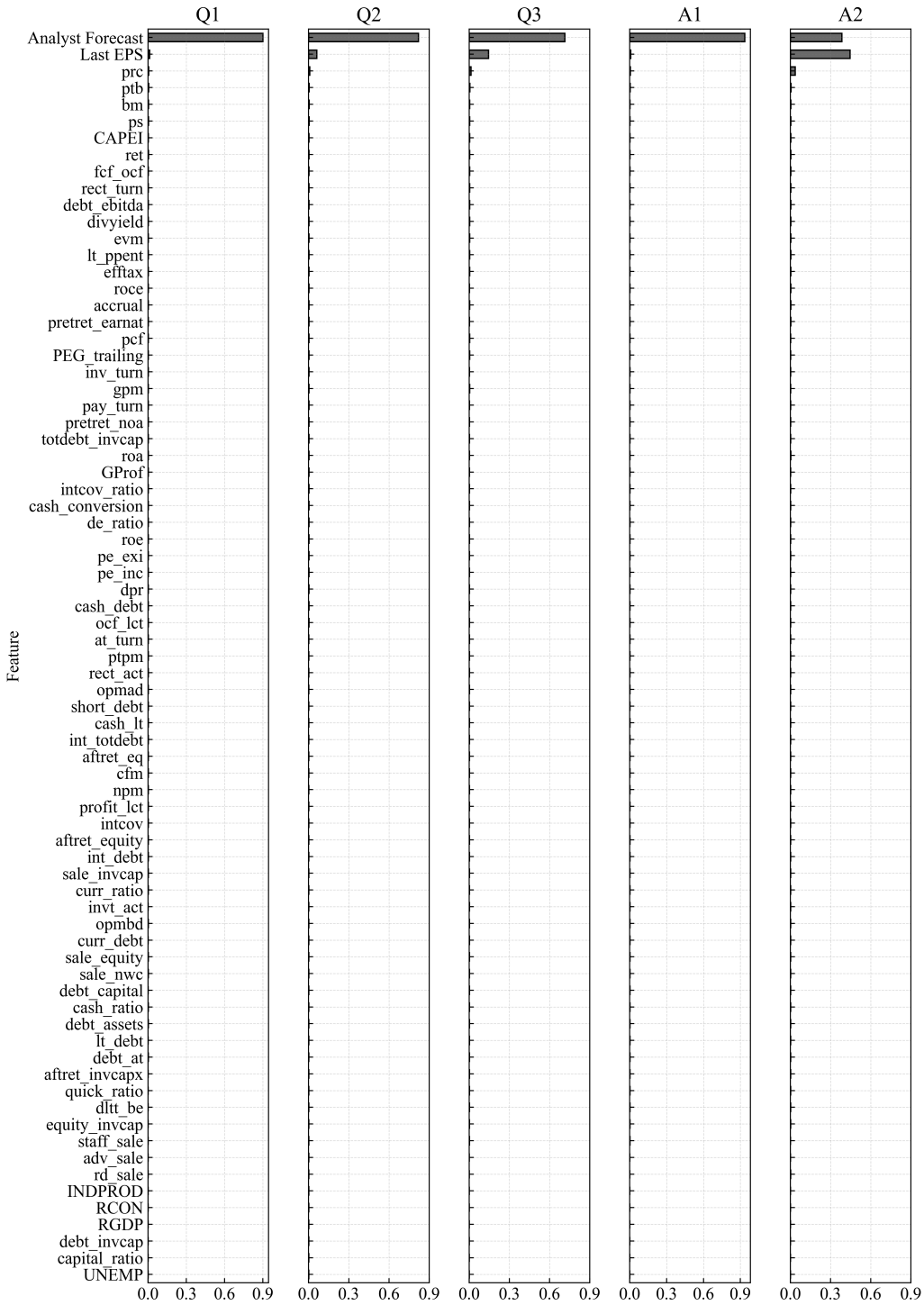


Figure 3: **Feature importance with look-ahead bias**. This figure plots the time-series average feature importance in random forest models trained with look-ahead bias using the baseline specification. Feature importance is computed as the average reduction of MSE when splitting by the feature, averaged across each node in a tree and then averaged across each tree. The feature importances are normalized to sum to one.

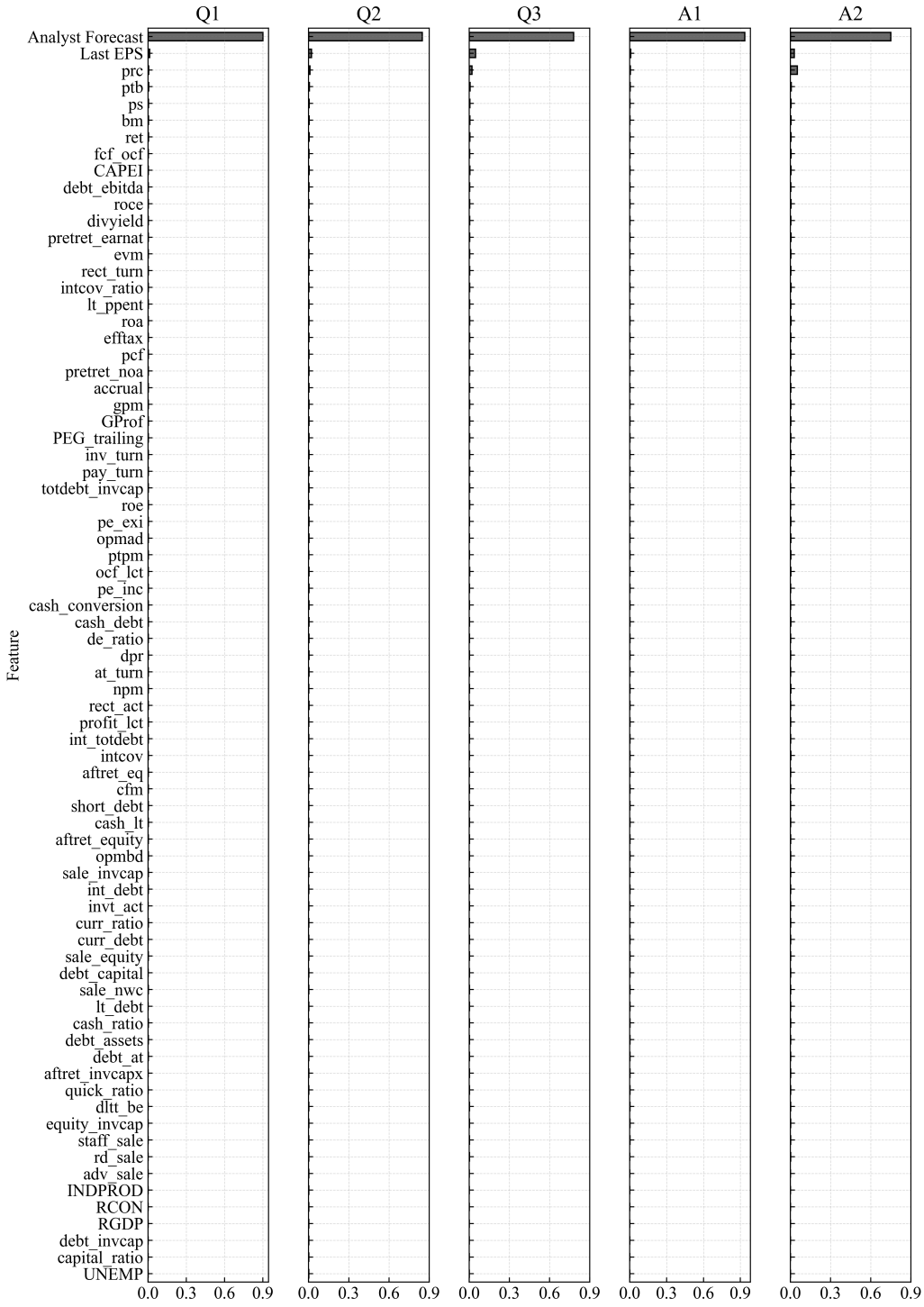


Figure 4: **Feature importance without look-ahead bias**. This figure plots the time-series average feature importance in random forest models trained without look-ahead bias.

References

Bianchi, Daniele, Matthias Büchner, Tobias Hoogteijling, and Andrea Tamoni, 2021, Corrigendum: Bond risk premiums with machine learning, *Review of Financial Studies* 34, 1090–1103.

Binsbergen, Jules H. van, Xiao Han, and Alejandro Lopez-Lira, 2023, Man versus machine learning: The term structure of earnings expectations and conditional biases, *Review of Financial Studies* 36, 2361–2396.

Bouchaud, Jean-Philippe, Philipp Krüger, Augustin Landier, and David Thesmar, 2019, Sticky expectations and the profitability anomaly, *Journal of Finance* 74, 639–674.

Breiman, Leo, 2001, Random forests, *Machine Learning* 45, 5–32.

Cakici, Nusret, Christian Fieberg, Tobias Neumaier, Thorsten Poddig, and Adam Zaremba, 2024, Pockets of predictability: A replication, *Journal of Finance* forthcoming.

Campbell, John L., Harrison Ham, Zhongjin Lu, and Katherine Wood, 2024, Expectations matter: When (not) to use machine learning earnings forecasts, working paper.

Cao, Sean, Norman Guo, Houping Xiao, and Baozhong Yang, 2024a, Can machines understand human skills? Insights from analyst selection, working paper.

Cao, Sean, Wei Jiang, Junbo Wang, and Baozhong Yang, 2024b, From man vs. machine to man + machine: The art and AI of stock analyses, *Journal of Financial Economics* forthcoming.

Cassella, Stefano, Benjamin Golez, Huseyin Gulen, and Peter Kelly, 2022, Horizon bias and the term structure of equity returns, *Review of Financial Studies* 36, 1253–1288.

Chen, Andrew Y., and Tom Zimmermann, 2022, Open source cross-sectional asset pricing, *Critical Finance Review* 11, 207–264.

Cooper, Michael J., Huseyin Gulen, and Michael J. Schill, 2008, Asset growth and the cross-section of stock returns, *Journal of Finance* 63, 1609–1651.

Dickerson, Alexander, Philippe Mueller, and Cesare Robotti, 2023, Priced risk in corporate bonds, *Journal of Financial Economics* 150, 103707.

Diether, Karl B., Christopher J. Malloy, and Anna Scherbina, 2002, Differences of opinion and the cross section of stock returns, *Journal of Finance* 57, 2113–2141.

Dong, Xi, Yan Li, David E. Rapach, and Guofu Zhou, 2022, Anomalies and the expected market return, *Journal of Finance* 77, 639–681.

Duarte, Jefferson, Christopher S. Jones, Mehdi Khorram, Haitao Mo, and Junbo Wang, 2023, Too good to be true: Look-ahead bias in empirical options research, working paper.

Engelberg, Joseph, R. David McLean, and Jeffrey Pontiff, 2020, Analysts and anomalies, *Journal of Accounting and Economics* 69, 101249.

Fama, Eugene F., and Kenneth R. French, 1993, Common risk factors in the returns on stocks and bonds, *Journal of Financial Economics* 33, 3–56.

Fama, Eugene F., and Kenneth R. French, 2015, A five-factor asset pricing model, *Journal of Financial Economics* 116, 1–22.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville, 2016, *Deep learning* (MIT Press).

Green, Jeremiah, John R.M. Hand, and X. Frank Zhang, 2017, The characteristics that provide independent information about average us monthly stock returns, *Review of Financial Studies* 30, 4389–4436.

Gu, Shihao, Bryan Kelly, and Dacheng Xiu, 2020, Empirical asset pricing via machine learning, *Review of Financial Studies* 33, 2223–2273.

Guenther, David A., and Andrew J. Rosman, 1994, Differences between COMPUSTAT and CRSP SIC codes and related effects on research, *Journal of Accounting and Economics* 18, 115–128.

Harvey, Campbell R., Yan Liu, and Heqing Zhu, 2016, ... and the cross-section of expected returns, *Review of Financial Studies* 29, 5–68.

Hastie, Trevor, Robert Tibshirani, Jerome H. Friedman, and Jerome H. Friedman, 2009, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, volume 2 (Springer).

Hou, Kewei, Chen Xue, and Lu Zhang, 2020, Replicating anomalies, *Review of Financial Studies* 33, 2019–2133.

Ke, Guolin, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu, 2017, LightGBM: A highly efficient gradient boosting decision tree, *Advances in Neural Information Processing Systems* 30.

Kothari, SP, Eric So, and Rodrigo Verdi, 2016, Analysts' forecasts and asset pricing: A survey, *Annual Review of Financial Economics* 8, 197–219.

La Porta, Rafael, 1996, Expectations and the cross-section of stock returns, *Journal of Finance* 51, 1715–1742.

Newey, Whitney K., and Kenneth D. West, 1987, A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix, *Econometrica* 55, 703–708.

Ross, Stephen A., 1976, The arbitrage theory of capital asset pricing, *Journal of Economic Theory* 13, 341–360.

Shumway, Tyler, 1997, The delisting bias in CRSP data, *Journal of Finance* 52, 327–340.

Silva, Tim de, and David Thesmar, 2024, Noise in expectations: Evidence from analyst forecasts, *Review of Financial Studies* 37, 1494–1537.

So, Eric C., 2013, A new approach to predicting analyst forecast errors: Do investors overweight analyst forecasts? *Journal of Financial Economics* 108, 615–640.

Tibshirani, Robert, 1996, Regression shrinkage and selection via the LASSO, *Journal of the Royal Statistical Society Series B: Statistical Methodology* 58, 267–288.

Verdineilli, Isabella, and Larry Wasserman, 2023, Feature importance: A closer look at Shapley values and LOCO, *arXiv preprint arXiv:* 2303.05981.

White, Halbert, 1980, A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity, *Econometrica* 817–838.

Wold, Herman, 1966, Estimation of principal components and related models by iterative least squares, in *Multivariate Analysis*, P.R. Krishnaiaah edition, 391–420 (Academic Press, New York).

Zhang, Yingguang, Guofu Zhou, and Yingzi Zhu, 2023, Myopic expectations and stock market mispricing, working paper.

Zou, Hui, and Trevor Hastie, 2005, Regularization and variable selection via the elastic net, *Journal of the Royal Statistical Society Series B: Statistical Methodology* 67, 301–320.

Table 3: The term structure of earnings forecasts via machine learning

	RF (1)	AF (2)	AE (3)	RF−AE (4)	t(RF−AE) (5)	AF−AE (6)	t(AF−AE) (7)	$(\text{RF}-\text{AE})^2$ (8)	$(\text{AF}-\text{AE})^2$ (9)	N (10)
Panel A: BHL original (Table 2)										
Q1	0.290	0.319	0.291	−0.000	−0.17	0.028	6.59	0.076	0.081	1,022,661
Q2	0.323	0.376	0.323	−0.001	−0.13	0.053	10.31	0.094	0.102	1,110,689
Q3	0.343	0.413	0.341	0.002	0.31	0.072	11.55	0.121	0.132	1,018,958
A1	1.194	1.320	1.167	0.027	1.64	0.154	6.24	0.670	0.686	1,260,060
A2	1.387	1.771	1.387	−0.004	−0.07	0.384	8.33	1.897	2.009	1,097,098
Panel B: Replication with look-ahead bias										
Q1	0.294	0.319	0.294	0.000	−0.14	0.026	6.50	0.065	0.070	1,010,411
Q2	0.324	0.376	0.327	−0.003	−0.65	0.049	10.28	0.082	0.090	1,106,291
Q3	0.344	0.412	0.344	−0.001	−0.13	0.068	11.44	0.105	0.120	1,016,839
A1	1.198	1.314	1.169	0.029	2.02	0.146	6.17	0.593	0.642	1,246,126
A2	1.362	1.731	1.367	−0.005	−0.10	0.364	8.28	1.628	1.837	1,126,934
Panel C: Replication without look-ahead bias										
Q1	0.294			0.000	−0.14			0.065		
Q2	**0.323**			**−0.003**	**−0.77**			**0.085**		
Q3	**0.343**			**−0.002**	**−0.28**			**0.114**		
A1	1.198			0.029	2.02			0.593		
A2	**1.368**			**0.001**	**0.02**			**1.788**		

This table reports time-series averages of machine learning EPS forecasts (RF), analysts' EPS forecasts (AF), and realized EPS (AE), and the time-series averages of the cross-sectional mean-squared forecasting errors by random forest, $(RF - AE)^2$, and by analysts, $(AF - AE)^2$. N denotes the number of observations. We also report the Newey and West (1987) adjusted t-values of the differences between earnings forecasts and realized earnings. The standard errors for quarterly forecasts are adjusted with three lags and those for annual forecasts are adjusted with twelve lags. Panel A reports estimates from Table 2 in BHL. Panels B and C show the replication results with (Panel B) and without (Panel C) the look-ahead bias. We omit the AF and AE numbers from Panel C because they would be identical to those in Panel B. The bolded estimates in Panel C are affected by the lookahead bias. The sample starts in January 1986 and ends in December 2019.

Table 4: Portfolios sorted on conditional bias

	Conditional bias (average BE) quintile					
	1	2	3	4	5	5−1
BHL original (Table 5)	1.32	0.98	0.79	0.47	−0.14	−1.46
	(6.53)	(4.53)	(3.18)	(1.62)	(−0.35)	(−5.11)
Replication with look-ahead bias	1.37	0.93	0.60	0.24	−0.17	−1.55
	(6.67)	(4.32)	(2.40)	(0.76)	(−0.42)	(−4.95)
Replication without look-ahead bias	1.01	0.89	0.93	0.80	0.74	−0.27
	(4.87)	(4.10)	(3.68)	(2.62)	(1.80)	(−0.86)

This table reports time-series average returns (in percent) for value-weighted monthly rebalanced portfolios formed on analysts' average conditional earnings forecast bias quintiles. The first and second rows are from Table 5 in BHL. The remaining four rows report our replication results with and without look-ahead bias. The t-values (in parentheses) are computed using White's (1980) heteroskedasticity robust standard errors as in BHL. The sample starts in January 1986 and ends in December 2019.

Table 5: Time-series tests with common asset pricing models

	CAPM		FF3		FF5	
	Coef.	t-value	Coef.	t-value	Coef.	t-value
Panel A: BHL original (Table 6)						
Intercept	−1.85	−7.18	−1.96	−8.64	−1.54	−5.84
MKT	0.56	7.53	0.53	7.86	0.38	5.28
SMB			0.80	7.06	0.61	5.17
HML			0.58	5.25	0.95	7.12
RMW					−0.68	−4.10
CMA					−0.53	−1.93
Panel B: Replication with look-ahead bias						
Intercept	−1.97	−7.15	−2.06	−8.51	−1.67	−5.95
MKT	0.61	7.65	0.57	7.68	0.43	5.81
SMB			0.89	7.32	0.66	5.17
HML			0.61	5.21	0.89	6.16
RMW					−0.75	−4.09
CMA					−0.30	−1.11
Panel C: Replication without look-ahead bias						
Intercept	−0.68	−2.52	−0.77	−3.32	−0.44	−1.63
MKT	0.60	7.78	0.54	7.74	0.43	5.86
SMB			0.97	8.02	0.76	5.85
HML			0.59	5.34	0.80	5.60
RMW					−0.70	−3.76
CMA					−0.15	−0.56

This table reports the alphas and factor loadings for the value-weighted long-short portfolio under the CAPM, Fama and French (1993) three-factor (FF3), and Fama and French (2015) five-factor models (FF5). The results in Panel A are from BHL's Table 6. Panels B and C present the results using our replicated average conditional bias (BE) with and without look-ahead bias. The t-values are computed using White's (1980) heteroskedasticity robust standard errors as in BHL. The sample starts in January 1986 and ends in December 2019.

Table 6: Diagnostic tests for look-ahead bias in earnings prediction

Dependent var.	h-period-ahead actual earnings: AE^h								
Horizon	$h = 1$ quarter			$h = 2$ quarters			$h = 1$ year		
RF^h	0.74			0.55			0.51		
	(11.71)			(8.25)			(11.19)		
RF^{h+1}	0.05			0.03			0.02		
	(1.25)			(0.71)			(1.38)		
BHL^h		−0.08			−0.24			−0.31	
		(−2.09)			(−5.37)			(−9.56)	
BHL^{h+1}		0.72			0.91			0.58	
		(13.22)			(19.73)			(30.12)	
RF^h_{LAB}			−0.21			−0.36			−0.39
			(−3.78)			(−7.66)			(−7.87)
RF^{h+1}_{LAB}			0.98			1.16			0.68
			(14.07)			(21.61)			(24.12)
AF^h	0.27	1.07	1.15	0.44	1.10	1.15	0.60	1.05	1.20
	(4.23)	(26.47)	(21.21)	(6.78)	(26.78)	(26.31)	(13.09)	(29.34)	(25.86)
AF^{h+1}	−0.05	−0.67	−0.89	−0.02	−0.74	−0.93	−0.12	−0.28	−0.47
	(−1.26)	(−12.99)	(−13.92)	(−0.55)	(−19.88)	(−20.95)	(−6.38)	(−18.25)	(−21.84)
Intercept	0.00	0.00	0.01	−0.01	−0.01	0.00	−0.04	−0.08	−0.01
	(−3.55)	(0.65)	(5.68)	(−6.79)	(−3.85)	(2.35)	(−5.09)	(−11.65)	(−1.75)
N	878,832	878,832	878,832	982,217	982,217	982,217	989,488	989,488	989,488
Adj. R^2	0.85	0.85	0.86	0.79	0.82	0.83	0.89	0.92	0.91

This table reports the results from regressions in which the dependent variables are the actual earnings h periods ahead and the explanatory variables are random forest forecasts for horizon h and $h+1$ earnings with no look-ahead bias (RF^h and RF^{h+1}), BHL's forecasts (BHL^h and BHL^{h+1}), and our replicated random forest forecasts for horizon h and $h+1$ with look-ahead bias (RF^h_{LAB} and RF^{h+1}_{LAB}). The control variables include analysts' forecasts for horizons h and $h+1$ (AF^h and AF^{h+1}). Horizon h takes the value of one quarter, two quarters, and one year. We report the t-values in paratheses. Standard errors are two-way clustered by month and stock.

Table 7: Average returns and alphas for long-short portfolios formed using a parsimonious OLS model

		Asset pricing model		
	High−Low	CAPM	FF3	FF5
Full-sample	−1.69	−2.13	−2.22	−1.81
	(−4.67)	(−6.71)	(−9.19)	(−6.34)
Pre-2000	−1.78	−1.95	−1.68	−1.59
	(−4.00)	(−3.90)	(−6.61)	(−5.90)
Post-2000	−1.62	−2.11	−2.32	−1.69
	(−3.06)	(−5.54)	(−8.57)	(−4.69)

This table reports average returns and alphas for a value-weighted monthly rebalanced long-short strategy that buys stocks in the top "Average BE" quintile and sells those in the bottom quintile. We report the estimates for the full, pre-2000, and post-2000 samples. We use the top five features from the random forest models each month to forecast future earnings at the one-quarter, two-quarter, three-quarter, one-year, and two-year horizons. We then compute the difference between analysts' and models' forecasts and average across all horizons as the average conditional bias. These strategies are not tradable in real time because one of the explanatory variables in the predictive model is "Realized earnings in the last period," which has a look-ahead bias. The t-values are Newey and West (1987) adjusted with 12 lags.

Table 8: Alternative random forests specifications

	Varying:	Sample fraction		Feature fraction		Training window		Standardization		Per-share
		Baseline = 0.05		1.0		1 year		None		No
	Baseline	0.01	0.10	$\sqrt{N_{\text{features}}}$	0.5	3 year	5 year	Rank	Z-score	Yes
Panel A: Mean-squared forecast error										
Q1	0.065	0.068	0.065	0.101	0.067	0.065	0.066	0.067	0.067	0.065
Q2	0.085	0.089	0.085	0.118	0.087	0.086	0.086	0.085	0.085	0.085
Q3	0.114	0.122	0.114	0.145	0.116	0.115	0.115	0.113	0.113	0.114
A1	0.593	0.689	0.590	1.017	0.632	0.591	0.591	0.621	0.618	0.594
A2	1.788	1.892	1.790	2.114	1.807	1.784	1.747	1.765	1.767	1.794
Panel B: Long-short portfolio performance										
Return	−0.27	−0.09	−0.25	0.17	−0.12	−0.25	−0.20	−0.20	−0.11	−0.23
	(−0.71)	(−0.27)	(−0.70)	(0.75)	(−0.35)	(−0.67)	(−0.60)	(−0.52)	(−0.31)	(−0.60)
CAPM	−0.68	−0.48	−0.67	0.06	−0.52	−0.70	−0.64	−0.53	−0.45	−0.66
	(−1.95)	(−1.55)	(−2.00)	(0.24)	(−1.61)	(−2.02)	(−2.09)	(−1.36)	(−1.24)	(−1.90)
FF3	−0.77	−0.54	−0.77	−0.09	−0.61	−0.79	−0.74	−0.70	−0.62	−0.75
	(−3.30)	(−2.49)	(−3.55)	(−0.45)	(−2.79)	(−3.43)	(−3.57)	(−2.63)	(−2.45)	(−3.14)
FF5	−0.44	−0.13	−0.42	0.09	−0.23	−0.43	−0.31	−0.51	−0.39	−0.39
	(−1.51)	(−0.44)	(−1.56)	(0.41)	(−0.84)	(−1.56)	(−1.19)	(−1.83)	(−1.27)	(−1.37)

This table reports on the performances of different implementations of random forest models. Panel A shows the mean-squared forecasting errors for each implementation for different forecasting horizons. Panel B reports on the performances of value-weighted monthly rebalanced quintile long-short portfolios formed on analysts' average conditional biases computed from the different models. We report the average returns and alphas from the CAPM and the Fama-French three- (FF3) and five-factor (FF5) models. The sample starts in January 1986 and ends in 2019. The Newey and West (1987) adjusted t-values with 12 lags are reported in parentheses.

Table 9: Alternative machine learning models

	OLS	PLS	LASSO	ENet	RF	LGBM	Stacking
Panel A: Mean-squared forecast error							
Q1	0.055	0.055	0.055	0.055	0.053	0.054	0.053
Q2	0.079	0.080	0.078	0.078	0.076	0.077	0.076
Q3	0.106	0.108	0.106	0.106	0.103	0.104	0.102
A1	0.548	0.545	0.547	0.546	0.540	0.548	0.525
A2	1.840	1.835	1.736	1.740	1.640	1.651	1.638
Panel B: Long-short portfolio performance							
Return	−0.24	−0.27	−0.23	−0.25	−0.02	−0.22	−0.23
	(−0.75)	(−0.83)	(−0.70)	(−0.76)	(−0.06)	(−0.62)	(−0.64)
CAPM	−0.67	−0.69	−0.63	−0.65	−0.47	−0.59	−0.67
	(−2.38)	(−2.49)	(−2.13)	(−2.25)	(−1.42)	(−1.78)	(−2.17)
FF3	−0.72	−0.73	−0.66	−0.68	−0.56	−0.69	−0.74
	(−3.48)	(−3.39)	(−3.15)	(−3.15)	(−2.21)	(−2.91)	(−3.43)
FF5	−0.28	−0.32	−0.28	−0.30	−0.13	−0.39	−0.34
	(−1.05)	(−1.15)	(−1.04)	(−1.09)	(−0.42)	(−1.36)	(−1.19)

This table reports on the performances of various machine learning models. Panel A shows the mean-squared forecasting errors for each model for different forecasting horizons. Panel B reports on the performances of value-weighted monthly rebalanced quintile long-short portfolios formed on analysts' average conditional biases computed from different models. We report the average returns and abnormal returns under the CAPM, Fama-French three-factor (FF3), and Fama-French five-factor (FF5) models. The sample starts in January 1989 and ends in 2019. The Newey and West (1987) adjusted t-values with 12 lags are reported in parentheses.

Appendix

A Supplementary Materials

Table A.1: Variable Definition

Variable	Definition	Variable	Definition
	A. Firm Fundamental		
	A.1 IBES actual (1)		
Last EPS	Last Quarter/Year Acutal EPS		
	A.2 WRDS Financial Ratios (67)		
accrual	Accruals/Average Assets	intcov_ratio	Interest Coverage Ratio
adv_sale	Avertising Expenses/Sales	inv_turn	Inventory Turnover
aftret_eq	After-tax Return on Average Common Equity	invt_act	Inventory/Current Assets
aftret_equity	After-tax Return on Total Stockholders Equity	lt_debt	Long-term Debt/Total Liabilities
aftret_invcapx	After-tax Return on Invested Capital	lt_ppent	Total Liabilities/Total Tangible Assets
at_turn	Asset Turnover	npm	Net Profit Margin
bm	Book/Market	ocf_lct	Operating CF/Current Liabilities
capei	Shillers Cyclically Adjusted P/E Ratio	opmad	Operating Profit Margin After Depreciation
capital_ratio	Capitalization Ratio	opmbd	Operating Profit Margin Before Depreciation
cash_conversion	Cash Conversion Cycle (Days)	pay_turn	Payables Turnover
cash_debt	Cash Flow/ Total Debt	pcf	Price/Cash fow
cash_lt	Cash Balance/Total Liabilities	pe_exi	P/E (Diluted, Excl. EI)
cash_ratio	Cash Ratio	pe_inc	P/E (Diluted, Incl. EI)
cfm	Cash Flow Margin	PEG_trailing	Trailing P/E to Growth (PEG) ratio
curr_debt	Current Liabilities/Total Liabilities	pretret_earnat	Pre-tax Return on Total Earning Assets
curr_ratio	Current Ratio	pretret_noa	Pre-tax return on Net Operating Assets
de_ratio	Total Debt/Equity	profit_lct	Proft Before Depreciation/Current Liabilities
debt_assets	Total Debt/Total Assets (1)	ps	Price/Sales
debt_at	Total Debt/Total Assets (2)	ptb	Price/Book
debt_capital	Total Debt/Capital	ptpm	Pre-tax Profit Margin
debt_ebitda	Total Debt/EBITDA	quick_ratio	Quick Ratio (Acid Test)
debt_invcap	Long-term Debt/Invested Capital	RD_SALE	Research and Development/Sales
divyield	Dividend Yield	rect_act	Receivables/Current Assets
dltt_be	Long-term Debt/Book Equity	rect_turn	Receivables Turnover
dpr	Dividend Payout Ratio	roa	Return on Assets

Continued on next page

Table A.1 – continued from previous page

Variable	Definition	Variable	Definition
efftax	Effective Tax Rate	roce	Return on Capital Employed
equity_invcap	Common Equity/ Invested Capital	roe	Return on Equity
evm	Enterprise Value Multiple	sale_equity	Sales/Stockholders Equity
fcf_ocf	Free Cash Flow/Operating Cash Flow	sale_invcap	Sales/Invested Capital
gpm	Gross Profit Margin	sale_nwc	Sales/Working Capital
GProf	Gross Profit/Total Assets	short_debt	Short-Term Debt/Total Debt
int_debt	Interest/ Average Long-term Debt	staff_sale	Labor Expenses/Sales
int_totdebt	Interest/Average Total Debt	totdebt_invcap	Total Debt/Invested Capital
intcov	After-tax Interest Coverage		
	A.3 CRSP (2)		
ret	Monthly Return	prc	Closing Price
	A.4 Fundamental Per Share (26, optional)		
dividend_p	Dividend Per Share	inventory_p	Inventory Per Share
BE_p	Book Equity Per Share	receivables_p	Receivables Per Share
Liability_p	Total Liability Per Share	Cur_debt_p	Short-term Debt Per Share
cur_liability_p	Current Liabilities Per Share	interest_p	Interest Per Share
LT_debt_p	Long-term Debt Per Share	fcf_ocf_p	Free Cash Flow Per Share
cash_p	Cash Balance Per Share	evm_p	Enterprise Value Per Share
total_asset_p	Total Asset Per Share	sales_p	Sales Per Share
tot_debt_p	Total Debt Per Share	invcap_p	Invested Capital Per Share
accrual_p	Accruals Per Share	c_equity_p	Current Equity Per Share
EBIT_p	EBITDA Per Share	rd_p	RD Per Share
cur_asset_p	Current Asset Per Share	opmad_p	Operating Income After Depreciation Per Share
pbda_p	Operating Income before DA Per Share	gpm_p	Gross Profit Per Share
ocf_p	Operating Cash Flow Per Share	ptpm_p	Pretax IncomePer Share
	B. Macroeconomic variables (4)		
RCON	Log Difference of Real Consumption	INDPROD	Log Difference of Industrial Production Index
RGDP	Log Difference of Real GDP	UNEMP	Unemployment rate
	C. Analyst forecasts (1)		
AF	Analyst Forecasts		

Table A.2: Different machine learning models: long-short portfolio performance by horizon

Panel A: Raw return

	OLS	PLS	LASSO	ENet	RF	LGBM	Stacking
Q1	−0.31	−0.33	−0.15	−0.21	−0.14	−0.28	−0.26
	(−0.89)	(−0.93)	(−0.42)	(−0.57)	(−0.41)	(−0.91)	(−0.73)
Q2	−0.36	−0.35	−0.48	−0.48	−0.24	−0.27	−0.31
	(−1.07)	(−1.03)	(−1.44)	(−1.51)	(−0.65)	(−0.82)	(−0.89)
Q3	−0.25	−0.26	−0.24	−0.24	0.02	−0.09	−0.25
	(−0.76)	(−0.82)	(−0.73)	(−0.73)	(0.05)	(−0.26)	(−0.72)
A1	−0.38	−0.31	−0.25	−0.30	−0.22	−0.24	−0.29
	(−1.16)	(−0.92)	(−0.78)	(−0.88)	(−0.60)	(−0.74)	(−0.83)
A2	0.01	−0.11	−0.26	−0.24	0.18	0.03	−0.08
	(0.02)	(−0.32)	(−0.76)	(−0.70)	(0.52)	(0.10)	(−0.25)

Panel B: Fama-French five-factor model alphas

	OLS	PLS	LASSO	ENet	RF	LGBM	Stacking
Q1	−0.31	−0.30	−0.13	−0.15	−0.12	−0.23	−0.29
	(−0.99)	(−0.95)	(−0.40)	(−0.47)	(−0.51)	(−0.86)	(−0.97)
Q2	−0.48	−0.50	−0.64	−0.68	−0.38	−0.38	−0.44
	(−1.61)	(−1.63)	(−2.46)	(−2.67)	(−1.30)	(−1.28)	(−1.46)
Q3	−0.33	−0.39	−0.37	−0.39	−0.17	−0.26	−0.43
	(−1.16)	(−1.41)	(−1.35)	(−1.47)	(−0.58)	(−0.91)	(−1.59)
A1	−0.37	−0.32	−0.29	−0.34	−0.34	−0.32	−0.34
	(−1.38)	(−1.13)	(−1.06)	(−1.24)	(−1.17)	(−1.14)	(−1.19)
A2	−0.12	−0.14	−0.27	−0.25	0.01	−0.07	−0.20
	(−0.48)	(−0.51)	(−0.93)	(−0.84)	(0.04)	(−0.24)	(−0.77)

This table reports long-short portfolio returns (Panel A) and Fama-French five-factor model alphas (Panel B) on conditional biases for different forecasting horizons and machine learning models. The alternative machine learning models are introduced in Section 4.2 and described in Appendix C. The sample starts in 1989 and ends in 2019. The Newey and West (1987) adjusted t-values with 12 lags are reported in parentheses.

Table A.3: Diagnostic tests for look-ahead bias in forecast error prediction

Dependent var.	Analysts' horizon-h forecast error: $AF^h - AE^h$								
Horizon	$h = 1$ quarter			$h = 2$ quarters			$h = 1$ year		
$AF^h - RF^h$	0.78			0.57			0.54		
	(13.56)			(8.99)			(12.40)		
$AF^{h+1} - RF^{h+1}$	0.03			0.02			0.03		
	(0.70)			(0.42)			(1.84)		
$AF^h - BHL^h$		0.02			−0.11			−0.27	
		(0.40)			(−2.35)			(−8.86)	
$AF^{h+1} - BHL^{h+1}$		0.65			0.78			0.47	
		(12.75)			(18.40)			(28.69)	
$AF^h - RF^h_{LAB}$			−0.02			−0.07			−0.32
			(−0.37)			(−1.44)			(−6.54)
$AF^{h+1} - RF^{h+1}_{LAB}$			0.84			0.89			0.58
			(13.71)			(19.88)			(23.93)
Intercept	0.00	−0.01	−0.02	0.01	0.00	−0.01	0.05	−0.04	−0.07
	(0.99)	(−9.75)	(−11.76)	(6.38)	(−1.62)	(−5.26)	(9.20)	(−6.53)	(−10.53)
N	878,832	878,832	878,832	982,217	982,217	982,217	989,488	989,488	989,488
Adj. R^2	0.06	0.08	0.12	0.04	0.16	0.18	0.03	0.25	0.22

This table reports the results from regressions in which the dependent variables are analysts' ex post forecast error at horizon h and the explanatory variables are analysts' bias implied by random forest forecasts for horizon h and $h+1$ with no look-ahead bias ($AF^h - RF^h$ and $AF^{h+1} - RF^{h+1}$), by BHL's forecasts ($AF^h - BHL^h$ and $AF^{h+1} - BHL^{h+1}$), and by our replicated random forest forecasts for horizon h and $h+1$ with look-ahead bias ($AF^h - RF^h_{LAB}$ and $AF^{h+1} - RF^{h+1}_{LAB}$). h takes the value of one quarter, two quarters, and one year. We report the t-values in paratheses. Standard errors are two-way clustered by time and stock.

Table A.4: Portfolios sorted on conditional bias: OLS forecasts with look-ahead bias

	Return	CAPM	FF3	FF5
Panel A: Top 5 features				
Full-sample	−1.69	−2.13	−2.22	−1.81
	(−4.67)	(−6.71)	(−9.19)	(−6.34)
Pre-2000	−1.78	−1.95	−1.68	−1.59
	(−4.00)	(−3.90)	(−6.61)	(−5.90)
Post-2000	−1.62	−2.11	−2.32	−1.69
	(−3.06)	(−5.54)	(−8.57)	(−4.69)
Panel B: Top 5 features (z-scores)				
Full-sample	−1.29	−1.61	−1.77	−1.52
	(−3.65)	(−4.89)	(−6.00)	(−4.03)
Pre-2000	−1.92	−2.13	−1.95	−1.88
	(−4.34)	(−4.57)	(−5.80)	(−5.19)
Post-2000	−0.84	−1.17	−1.34	−0.85
	(−1.73)	(−2.72)	(−3.52)	(−1.77)
Panel C: All 75 features				
Full-sample	−1.53	−1.93	−2.01	−1.67
	(−4.46)	(−6.19)	(−7.99)	(−5.56)
Pre-2000	−1.73	−1.92	−1.69	−1.61
	(−3.87)	(−3.92)	(−5.83)	(−5.88)
Post-2000	−1.39	−1.82	−1.99	−1.46
	(−2.83)	(−4.65)	(−6.38)	(−3.65)
Panel D: All 75 features (z-scores)				
Full-sample	−1.17	−1.46	−1.65	−1.44
	(−3.19)	(−4.09)	(−5.49)	(−3.82)
Pre-2000	−1.93	−2.17	−2.00	−1.93
	(−4.32)	(−4.63)	(−5.35)	(−5.20)
Post-2000	−0.63	−0.92	−1.10	−0.63
	(−1.21)	(−1.81)	(−2.62)	(−1.20)

This table reports on the performances of value-weighted monthly rebalanced long-short portfolios that buy stocks in the top average conditional bias quintile and sell those in the bottom quintile. The conditional biases are estimated with a look-ahead bias. We report the average return, the CAPM, and the Fama-French three-factor (FF3) and five-factor (FF5) model alphas and their t-values (in parentheses) for the full-sample, pre-2000, and post-2000 samples. The OLS models for estimating conditional bias in Panels A, B, C, and D differ by the number of features (top 5 or all 75) and by the use of raw or z-score features. The t-values are Newey and West (1987) adjusted with 12 lags.

Table A.5: 56 random forest specifications and portfolio performance

ID	Sample	Feature	Window	Standardization	Per-share	FF5 Alpha	t-value
1	0.10	1	12	Rank	T	−0.55	−2.13
2	0.05	1	12	Rank	T	−0.51	−1.93
3	0.05	1	12	Rank	F	−0.51	−1.83
4	0.10	1	12	Rank	F	−0.48	−1.79
5	0.05	1	48	Z-score	T	−0.51	−1.76
6	0.10	1	24	Rank	T	−0.43	−1.70
7	0.10	1	24	None	F	−0.45	−1.67
8	0.10	1	12	None	F	−0.42	−1.56
9	0.05	1	24	Rank	F	−0.40	−1.56
10	0.05	1	36	None	F	−0.43	−1.56
11	0.05	1	24	Rank	T	−0.39	−1.54
12	0.05	1	12	None	F	−0.44	−1.51
13	0.10	1	24	Rank	F	−0.38	−1.51
14	0.05	1	24	None	F	−0.42	−1.49
15	0.10	1	12	Z-score	F	−0.46	−1.47
16	0.10	1	24	Z-score	F	−0.44	−1.45
17	0.10	1	12	None	T	−0.41	−1.45
18	0.05	1	24	Z-score	F	−0.43	−1.43
19	0.05	1	12	None	T	−0.39	−1.38
20	0.05	1	24	Z-score	T	−0.42	−1.37
21	0.10	1	24	Z-score	T	−0.41	−1.35
22	0.10	1	24	None	T	−0.36	−1.33
23	0.05	1	12	Z-score	T	−0.43	−1.32
24	0.05	1	12	Z-score	F	−0.39	−1.27
25	0.05	1	24	None	T	−0.34	−1.21
26	0.05	1	60	None	F	−0.31	−1.19
27	0.10	1	12	Z-score	T	−0.37	−1.12
28	0.05	$\sqrt{N}$	48	Rank	T	−0.31	−1.11
29	0.10	$\sqrt{N}$	48	Rank	T	−0.31	−1.11
30	0.10	$\sqrt{N}$	48	Z-score	T	−0.30	−1.10
31	0.05	$\sqrt{N}$	48	Z-score	T	−0.30	−1.09
32	0.10	$\sqrt{N}$	12	Z-score	T	−0.26	−0.99
33	0.05	$\sqrt{N}$	12	Z-score	T	−0.25	−0.98
34	0.05	$\sqrt{N}$	24	Z-score	T	−0.23	−0.90
35	0.05	$\sqrt{N}$	12	Rank	T	−0.21	−0.89
36	0.10	$\sqrt{N}$	24	Rank	T	−0.22	−0.87
37	0.05	$\sqrt{N}$	24	Rank	T	−0.21	−0.84
38	0.10	$\sqrt{N}$	24	Z-score	T	−0.20	−0.76
39	0.10	$\sqrt{N}$	12	Rank	T	−0.18	−0.76
40	0.10	$\sqrt{N}$	12	None	T	−0.15	−0.71
41	0.10	$\sqrt{N}$	24	Rank	F	−0.16	−0.62
42	0.10	$\sqrt{N}$	24	Z-score	F	−0.13	−0.56
43	0.05	$\sqrt{N}$	24	Rank	F	−0.12	−0.48
44	0.10	$\sqrt{N}$	12	Rank	F	−0.11	−0.48
45	0.05	$\sqrt{N}$	12	Rank	F	−0.10	−0.44
46	0.01	1	12	None	F	−0.13	−0.44
47	0.05	$\sqrt{N}$	24	Z-score	F	−0.10	−0.42
48	0.05	$\sqrt{N}$	12	Z-score	F	−0.10	−0.42
49	0.10	$\sqrt{N}$	12	Z-score	F	−0.09	−0.40
50	0.10	$\sqrt{N}$	24	None	T	−0.05	−0.20
51	0.05	$\sqrt{N}$	12	None	T	−0.03	−0.16
52	0.05	$\sqrt{N}$	24	None	T	−0.02	−0.07
53	0.10	$\sqrt{N}$	12	None	F	0.07	0.32
54	0.10	$\sqrt{N}$	24	None	F	0.08	0.36
55	0.05	$\sqrt{N}$	12	None	F	0.09	0.41
56	0.05	$\sqrt{N}$	24	None	F	0.10	0.43

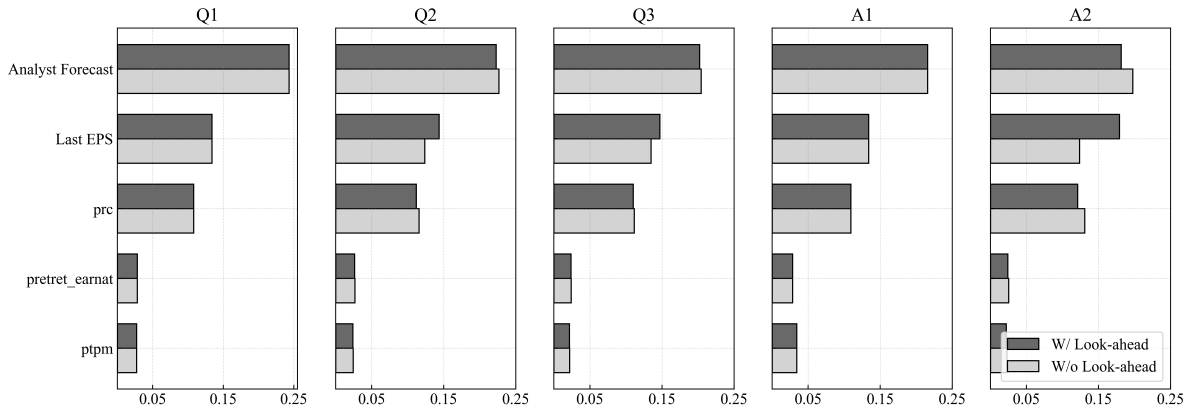


Figure A.1: **Feature importance in random forest models using $\sqrt{\mathbf{N_{features}}}$ features**. This figure plots the time-series average of feature importance of the 5 most important features with and without look-ahead bias. The feature importance is the reduction in mean-squared error when splitting on the feature averaged across each node within a tree and then averaged across each tree. The feature importance for each variable is normalized to sum to one. The number of random features considered by each split equals the square root (rounded down) of the total number of features, that is, $\lfloor\sqrt{75}\rfloor = 8$.

B Sample Construction

We outline the sample construction steps below:

1. **Merging IBES consensus forecasts with IBES actual data**. We match the IBES actual earnings and consensus earnings forecast data using the IBES ticker and fiscal period end date. The fiscal period end date is named PENDS in the actual file and FPEDATS in the summary forecast file.
2. **Merging CRSP data**. We match IBES and CRSP data via the link table provided by the Wharton Research Data Service. We also require firms' historical CUSIP to be the same in both IBES and CRSP (CUSIP in IBES and NCUSIP in CRSP).
3. **Delisting return**. When missing, we set delisting returns to -50% (Shumway, 1997).
4. **Adjusting IBES Earnings Per Share**. We use the IBES Unadjusted Files to prevent rounding errors found in the IBES Adjusted Files (Diether et al., 2002). We adjust for dividends and share splits using CRSP's cumulative adjustment factor (CFACSHR) to maintain consistency in the share base between forecasts and actual earnings. Specifically, if forecasts are made at $t-1$ and the actual earnings are announced at t, we compute the adjusted actual earning as,

$$AdjustActual_t = Actual_t \times CFACSHR_{t-1}/CFACSHR_t.$$

5. **Merging WRDS Financial Ratios Data**. To identify a firm's industry, we first match WRDS Financial Ratio data with Compustat using the GVKEY identifier. The Standard Industrial Classification (SIC) code is primarily sourced from Compustat (Guenther and Rosman, 1994). In cases where the SIC code is unavailable, we use the SIC code from CRSP. We match WRDS financial ratios with our data using the PERMNO identifier.

C Machine Learning Models

In this section, we provide a brief overview of the machine learning models employed. For a more detailed introduction, we refer the reader to Hastie et al. (2009). Given a dataset (y, X), where y is the target variable and X is a set of features, the goal of machine learning models is to find a function $f(\cdot)$ that best fits the data by some criteria, or formally, minimizing the loss function $\mathcal{L}(y, f(X))$. In our context, the target variable is the actual earnings per share for different horizons. The features X are comprised of firms' characteristics, macroeconomic variables, and analysts' forecasts.

C.1 Linear regressions

Ordinary least squares regression (OLS) imposes that the function $f(\cdot)$ is a linear function of the predictor variables X and the parameter vector β,

$$f(X) = X\beta, \tag{A.1}$$

where X is a $N \times P$ matrix and N and P denote the number of observations and predictors, respectively. The training goal is to find the parameters β^* that minimize the L2 loss:

$$\beta^* = \arg\min_{\beta} \|y - X\beta\|_2. \tag{A.2}$$

C.1.1 LASSO

Penalized least squares regressions modify the OLS loss function by adding a penalty term $\phi(\beta)$:

$$\mathcal{L} = \|y - X\beta\|_2 + \phi(\beta). \tag{A.3}$$

LASSO (Tibshirani, 1996, Least Absolute Shrinkage and Selection Operator) regression imposes a

L1 penalty on the size of the coefficients:

$$\phi(\beta) = \lambda \sum_{j=1}^{P} |\beta_j|. \tag{A.4}$$

The key characteristic of LASSO regression is its ability to perform variable selection by driving the coefficients on less important variables to zero. Therefore, it is useful in situations where the dataset includes a large number of predictors.

C.1.2 Elastic net

Elastic net (Zou and Hastie, 2005, ENet) includes both a L1 penalty, which can select variables, and a L2 penalty term, which helps handle multicollinearity:

$$\phi(\beta; \lambda, \rho) = \lambda(1-\rho) \sum_{j=1}^{P} |\beta_j| + \frac{1}{2}\lambda\rho \sum_{j=1}^{P} \beta_j^2, \tag{A.5}$$

where λ is a tuning parameter that controls the overall strength of the penalty and ρ adjusts the balance between the LASSO (L1) and Ridge (L2) penalties.

C.2 Partial least squares

Partial Least Squares (PLS) regression (Wold, 1966) operates by constructing new predictor variables, known as latent variables, which are linear combinations of the original predictors. These latent variables are constructed iteratively, each one chosen to maximize the explained variance of the dependent variable.

PLS works as follows. For each predictor x_j, the algorithm estimates its univariate regression coefficient ϕ_j for explaining y. This coefficient indicates the "partial" sensitivity of target variable y to each predictor x_j. Next, average all predictors into a single aggregate component with weights proportional to ϕ_j. In this way, PLS performs its dimension reduction with the ultimate forecasting objective in mind.

To form more than one predictive component, the target variable and all predictors are orthog-

onalized against the components already developed. The aforementioned steps are then repeated on this orthogonalized dataset. This process continues iteratively until the predefined number of PLS components, denoted as K, is reached.

C.3 Tree-based models

Tree-based models use a series of decision rules derived from the features to make predictions. At the core of tree-based methods is the decision tree, which adaptively partition the dataset into a number of simple regions. Formally, a decision tree can be represented as a function:

$$f(X) = \sum_{m=1}^{M} c_m \cdot I(x \in R_m), \tag{A.6}$$

where M is the number of leaves in the tree, c_m is the value predicted in the mth region (leaf), R_m is the region corresponding to the mth leaf, and I is an indicator function that is 1 if x falls into the region R_m and 0 otherwise.

Regression trees are invariant to monotonic transformations of predictors and are designed to capture nonlinearity and interactions among predictors. However, the primary challenge in using decision trees is their tendency to overfit, especially with complex trees. Therefore, we use two "ensemble" tree models that combine forecasts from many different trees into a single forecast: the random forests and gradient boosted regression trees.

C.3.1 Random forests

Random forests, as introduced by Breiman (2001), involves constructing a multitude of decision trees to form a robust ensemble model. The process begins by generating multiple bootstrap samples from the original dataset. Each sample is used to train a deep decision tree independently. Unlike conventional decision trees, random forests also enhance diversity in the model by randomly selecting a subset of features for splitting nodes in each tree. This feature selection reduces the correlation between individual trees. The final prediction of the random forest is given by the

average of the predictions of all trees:

$$f(X) = \frac{1}{B} \sum_{b=1}^{B} f_b(X), \tag{A.7}$$

where B represents the number of trees, and $f_b(X)$ is the prediction of the bth tree. The key tuning parameters for this model include the number of trees in the ensemble, the depth of individual trees, the fraction of bootstrapped sample, and the fraction of features considered for splitting at each node.

C.3.2 Boosting

Gradient boosting trees is another ensemble technique that builds trees in a sequential manner, where each new tree is created to fit the residuals and correct the errors made by previously built trees. Unlike random forests, which build each tree independently, gradient boosting methods build one tree at a time. At each stage b, a new tree f_b is fit on the negative gradient of the loss function $\mathcal{L}$ used, which is typically the squared error for regression.

The model after B trees is given by:

$$F_B(X) = F_{B-1}(X) + \nu \cdot f_B(X), \tag{A.8}$$

where F_{B-1} is the model built using the previous $B-1$ trees, f_B is the new tree, and ν is the learning rate, a small positive number that scales the contribution of each tree.

Each tree in the gradient boosting framework is typically shallow, and this shallow nature helps keep the model robust and generalizable. Gradient boosting models are often tuned through several hyperparameters, including the number of trees, depth of trees, and learning rate, which control the complexity and speed of the learning process. We employ LightGBM (Ke et al., 2017), a highly efficient gradient boosting framework developed by Microsoft, recognized for its performance across various applications.[27]

[27]For further details, see the machine learning challenge-winning solutions using LightGBM at https://github.com/microsoft/LightGBM/tree/master/examples.

C.4 Hyperparameters

Table A.6 describes the set of hyperparameters and their potential values used for tuning each machine learning model.

Table A.6: Hyperparameter Grids for Machine Learning Models

OLS	PLS	LASSO	ENet	RF	LGBM
	$K \in [1, 20]$	$\lambda \in (10^{-4}, 10^{-1})$	$\rho = 0.5$ $\lambda \in (10^{-4}, 10^{-1})$	Depth= $4 \sim 10$ Trees= 1000 Sample Fraction $= 0.05$	Depth= $3 \sim 6$ Learning Rate $\in \{0.01, 0.03, .07, 0.1\}$ Trees= $\{100, 200, 300\}$

This table describes the hyperparameters that we tune in each machine learning method.