Dividend Forecasts via Machine Learning

Xuesi Wang[a], Leonidas G. Barbopoulos[a], and Khaladdin Rzayev[a,b]

[a]*University of Edinburgh, Business School, Edinburgh, UK*
[b]*Koc University, Istanbul, Turkey*

Abstract

This paper introduces real-time dividend forecasts using machine learning. We present evidence that our forecasts are associated with lower forecast errors compared to predictions produced with alternative methods. We also show that while analysts produce less accurate forecasts when dealing with firms with more complex information environments, our forecasts are not impacted by firm complexity, underscoring the effectiveness of machine-driven models in handling informationally challenging structures. Finally, we employ machine learning-based dividend forecasts to calculate firm-level expected returns and document that our measures outperform alternative proxies, both in-sample and out-of-sample.

Keywords: Dividend Forecast; Machine Learning; Expected Return; Payout Policy; Firm Complexity.

JEL codes: G11, G12, G14.

1. Introduction

Recent advancements in financial technology and machine learning, together with access to large volumes of data (known as *big data*), have fundamentally reshaped information production dynamics, thereby influencing economic decision-making and economic growth (Chalfin et al., 2016; Dugast and Foucault, 2023; Dessaint et al., 2023). In this study, we examine how these advancements affect the demand and accuracy of dividend forecasts. One possibility is that these innovations enhance the precision of dividend forecasts by leveraging richer firm-level information production dynamics. Yet, another possibility is that modern machine learning methods accessing large-and-diverse data may add more noise and distortions (Dugast and Foucault, 2018), potentially deteriorating the dividend forecasting accuracy.

Determining which of these opposing effects dominates remains an important empirical question that has broad implications for the efficient allocation of resources and risk. *First*, Allen et al. (2000) demonstrate that certain institutional investors, such as pension funds, consider dividend-paying firms as more valuable due to tax-related benefits accrued to investors. Therefore, the amount of dividends a firm distributes to its shareholders can directly influence the capital allocation and risk management decisions of both investors and portfolio managers. Anecdotal evidence strengthens the economic importance of this phenomenon is the recent surge in the demand for dividend forecasts. Figure 1 illustrates that the proportion of all dividend-paying firms, whose dividends forecasts are produced by analysts, increased from 3% in 2001 to roughly 90% in 2021. Similarly, the average number of analysts generating dividend forecasts has increased from 1 to 5 during the same period. This rise in demand for dividend forecasts is further highlighted with the significant holdings of dividend-paying stocks, suggesting that information demand preceding important events can act as a proxy for investor uncertainty and influence the impact of these events in financial markets (Benamar et al., 2021).

[Figure 1 about here.]

Second, accurate dividend forecasts have important implications for asset pricing. The expected dividend is an essential input for stock valuation methods such as Dividend Discount Models, hence more accurate dividend forecasts can enhance the precision of stock price estimations. Moreover, Kane et al. (1984) and Ely and Mande (1996) suggest that investors use information on earnings and dividends in tandem, impacting stock prices through their interaction. Thus, information originating

from dividend forecasts remains a significant input to asset pricing models, even after considering earnings forecasts.

Third, investigating the usefulness of machine learning to produce dividend forecasts can shed light on market efficiency. If machine learning approaches consistently provide more accurate dividend forecasts compared to what is already reflected in security prices, it suggests that the market may not be perfectly efficient in incorporating all available dividend-related information in stock prices. Along these lines, the ability of machine learning models to consistently outperform market prices in dividend forecasting may imply that there is room for improvement in how the market processes and responds to dividend-related information.

Motivated by these important considerations, we make a contribution to the existing literature on dividend forecasting. Specifically, we explore the potential of machine learning techniques to produce dividend forecasts with reduced biases and increased accuracy. Along theses studies, prior studies propose three main approaches to dividend forecasting: (i) historical realized dividend payouts, (ii) earnings forecasts multiplied by the payout ratio and, (iii) analysts' dividend forecasts.

The first two approaches that are discussed by Lintner (1956) and Brav et al. (2005) are based on three assumptions that are unlikely to hold in most situations. Historical dividends, expected earnings, and the payout ratio are assumed to contain sufficient information to predict future dividends accurately. In addition, a linear relationship between expected dividends and their determinants is assumed. Lastly, these methods rely on a stable, long-term payout ratio. However, there is limited empirical support for these assumptions and, by relying solely on them for dividend forecasting can introduce several biases, especially when considering that payout ratios are often estimated rather than directly observed. As the Figure (2a) demonstrates, over the period from 2001 to 2021, approximately 50% of dividend-paying firms maintained a constant Dividend-Per-Share (DPS) compared to the previous fiscal year. Additionally, a rather significant fraction of firms either increased or decreased their dividend payouts relative to the preceding year. Moreover, Figure (2b) also suggests that dividend payout ratio is also non-stable for a substantial fraction of firms. Therefore, this empirical evidence highlights the substantial bias of dividend forecast produced from purely sticky dividend assumption or previous target payout ratios.

[Figure 2 about here.]

The third approach, which is relying on analysts' dividend forecasts, has received significant attention in recent years. De La O and Myers (2021) use dividend forecasts from I/B/E/S to highlight

the relative importance of subjective cash flow news compared to discount rate news. Bilinski and Bradshaw (2022) further investigate the informativeness and accuracy of analyst forecasts for dividends, and show that they are indeed informative (Francis and Philbrick, 1993; Bartov et al., 2002). However, Hoitash et al. (2021) and Barinov et al. (2022) show that the accuracy and information content of analyst forecasts decline when analysts provide forecasts for firms that operate in opaque and complex information environments. As a result, a natural question that emerges is whether it is possible to obtain more accurate dividend forecasts by relying on a broader set of information and leveraging advanced machine learning techniques?

Our answer to this question is, in short, yes. Based on the Mean Squared Error (MSE) decomposition framework developed by de Silva and Thesmar (2021), we begin by theoretically showing that combining dividend forecasts using both analyst consensus and forecasts generated from publicly available general information, can outperform dividend forecasts solely based on Lintner's model and analyst consensus. Then, we leverage a comprehensive dataset that includes analyst predictions, financial statements, stock market data, and macroeconomic indicators, by employing three tree-based supervised learning regression models: (i) random forest, (ii) gradient boost trees, and (iii) extreme-gradient boost trees to generate dividend forecasts. Our approach uses a rolling-window estimation to construct the conditional dividend expectations. This permits data-driven determination of variable influences on dividend payouts and the evolution of these relationships over time without the need for predefined implicit assumptions in forecasting models. Tree-based ensemble algorithms offer two key advantages, compared to traditional methods, similar to linear forecasting when constructing dividend forecasts.

One of the advantages of machine learning algorithms is that they allow for a nonlinear relationship between explanatory and forecasted variables, a feature often overlooked by traditional econometric forecasting models (Gu et al., 2020; van Binsbergen et al., 2022). Moreover, the tree-based ensemble algorithm ensures an unbiased forecast, even with a large number of explanatory variables. This is simply achieved by leveraging all available (relevant) information to estimate the expected dividend. This aspect addresses a common challenge in econometric modeling, as noted by Kelly et al. (2022), where traditional OLS estimators are likley to become unreliable with an abundance of explanatory variables.[1]

[1]While non-parametric random forest regression approaches facilitate data-driven forecasting and allow us to include as many variables as possible, our selection of relevant variables is grounded in both theoretical and empirical studies that provide insights into dividend payout determinants. This intentional selection, rather than arbitrary feature inclusion, is guided by the recognition that dividend payout is fundamentally influenced by the payout policy established by a firm's

We show that our machine-learning dividend forecasts outperform those made by analysts, as well as forecasts based solely on historical dividends or those derived from the products of earnings forecasts and payout ratios in terms of forecasting-accuracy. The superiority of machine learning forecasts is particularly evident in long-term horizons. These results indicate that machine learning techniques effectively analyze a broader set of information compared to alternative methods. Consistent with this insight, we find that analysts produce less accurate dividend forecasts for firms with more complex information structures, whereas machine learning maintains its high level of accuracy. This underscores the economic advantages of using machine learning in forecasting, especially for firms with complex information structures.

As an application, we use our machine learning-based dividend forecasts to calculate expected returns based on the Dividend Discount Model, commonly referred to as the implied cost of capital (ICC) in the literature. Subsequently, we conduct a "horse race" to compare our expected returns against benchmark measures. These benchmarks are derived from forecasts of earnings, both from analysts and machine learning models, multiplied by one minus an estimated plowback rate (Pástor et al., 2008; Li et al., 2013).

During the period from January 2005 to December 2021, we assess the time-series predictability of equal-weighted and value-weighted monthly returns on the overall market, represented by the S&P 500 index. Our results suggest that our dividend-based expected return outperforms those produced using earnings multiplied by the estimated payout ratio. The higher predictability of returns remains robust when compared to other, widely used valuation ratios and business cycle indicators, both in- and out-of-sample, as well as when employing a portfolio-based approach.

We contribute to two strands of studies in the literature. First, our study aligns with the prevailing trend of employing machine learning techniques to shape expectations regarding a firm's future fundamentals, with a particular emphasis on earnings. van Binsbergen et al. (2022) employ random forests to address nonlinearity in earnings forecasts, thereby establishing a real-time, statistically optimal benchmark for firm-level earnings expectations. Moreover, de Silva and Thesmar (2021) offer a detailed analysis of earnings forecast accuracy, highlighting how machine learning techniques can mitigate the bias and noise often associated with analyst forecasts or traditional econometric methods.

management. Therefore, forming accurate expectations necessitates tracking the dynamics of relevant information mirroring managerial decision-making. For example, we incorporate specific firm actions such as mergers and acquisitions (Zwiebel, 1996), tax payouts (Yagan, 2015), share repurchases (Floyd et al., 2015), and the valuation of investment projects (Miller and Rock, 1985) into our predictors (please refer to the Internet Appendix Table B.1 for a comprehensive list of input predictors). Consequently, the feature importance derived from the tree-based regression models also serves to validate the significance of these firm actions in determining dividend payouts.

While the application of machine learning approaches to estimate earnings expectations has gained substantial attention in recent financial economics research, the domain of dividends has been relatively underexplored. Our research seeks to bridge this gap, by also emphasizing the importance of dividend forecasting. This emphasis is particularly relevant nowadays as Michaely and Moin (2022) demonstrate that since the early 2000s, the proportion of dividend-paying firms has notably increased. This trend suggests that investors are increasingly interested in cash dividend payments as a return on their asset holdings, underscoring an increased demand for dividend forecasts (Bilinski and Bradshaw, 2022). The real-time and statistically optimal dividend payout forecast that we provide in this study, using a machine-learning approach, can serve as a valuable benchmark for investors to consider, complementing predictions derived from analyst forecasts or traditional linear models.

Our study also contributes to the aforementioned papers by shedding light on one of the underlying reasons that enable machine learning to produce more accurate forecasts. Specifically, we find that machine learning-based dividend forecasts are less influenced by firm complexity compared to analysts' forecasts, showcasing the capacity of machines to handle more comprehensive information. This result implies that the economic advantages of employing machine learning algorithms in forecasting fundamentals are more pronounced in dealing with complex information environments. As the data generated by various business stakeholders continues to expand exponentially, expected to reach 175 zetabytes by 2025,[2] we anticipate an increasing role for machine learning techniques in the future.

Lastly, developing a forward-looking return measure has long been a central theme in financial economics. In our second contribution to the financial economics literature, we propose an expected return measure derived from the Dividend Discount Model, which incorporates our machine learning-based dividend forecasts as a critical input. Through a comparison with traditional benchmarks (Pástor et al., 2008; Li et al., 2013; Lee et al., 2020), we demonstrate that our expected return measure provides superior predictive ability about future realized returns. It is worth highlighting that while calculating the new expected return is a valuable application of our machine learning-based dividend forecasts, the primary focus of our contribution lies in the dividend forecast measure itself.

The remaining of the paper is structured as follows: Section (2) offers theoretical insights into the challenges posed by arbitrarily estimated payout ratios in calculating dividend forecasts. Sec-

[2] https://www2.deloitte.com/cy/en/pages/technology/articles/data-grown-big-value.html

tion (3) provides an overview of the data and empirical methodology used to construct our machine learning dividend expectations and presents the corresponding results. Section (4) explores how firm complexity impacts the accuracy of dividend forecasts. Section (5) discusses the calculation of our expected return measure and examines its empirical performance. Finally, Section (6) provides the paper's conclusion.

2. Mean Square Error decomposition of dividend expectations

We begin by revisiting the theoretical foundation underpinning existing dividend expectations. We then expand upon the decomposition of the forecast mean squared error (MSE) as formulated by de Silva and Thesmar (2021). Through this, we aim to illustrate how an optimal forecast, which integrates both analyst predictions and forecasts grounded in publicly available information, can enhance forecast efficiency.

2.1. Dividend expectation based on Lintner's model

Lintner (1956) proposes the following data generation process (DGP) that characterizes a firm's payout policy:

$$\Delta D_{t+1} = \gamma(D^*_{t+1} - D_t), \tag{1}$$

where ΔD_{t+1} represents the change in dividends from time t to $t+1$, with D_t denoting the dividend at time t. D^*_{t+1} represents the target dividend payout, calculated as the product of earnings E_{t+1} and a long-term target payout ratio θ_{t+1}.[3] The parameter γ represents the estimated speed of adjustment (SoA), a value varying from 0 to 1.

We rewrite Eq. (1) by taking conditional expectations on both sides:

$$\mathbb{E}_t(D_{t+1}) = (1-\gamma)D_t + \gamma\mathbb{E}_t(E_{t+1})\mathbb{E}_t(\theta_{t+1}). \tag{2}$$

Equation (2) provides several valuable insights. It reveals that the expectation of future dividends is a linear function of (i) the current dividend level, (ii) expected future earnings, (iii) the expected payout ratio and, (iv) the SoA that is implicit in the payout policy. This equation sheds light on firms' dividend payout policies from two distinct perspectives. First, it suggests that firms tend to

[3] In Lintner's original model, θ is typically considered a constant. However, in practice, θ is often estimated using historical payout ratios to mitigate small-sample bias, as discussed by Leary and Michaely (2011). Hence, we treat θ as time-varying in this context.

maintain a consistent dividend payout level over time. Second, it highlights that a firm's dividend payout is closely linked to its expected earnings and payout ratio. When $\gamma = 0$, dividend smoothing is maximized, and the firm is expected to maintain a stable dividend equivalent to its current payout level. By contrast, with γ approaching 1 the firm is expected to adjust its dividend solely based on expected earnings and the payout ratio, allowing for greater flexibility in dividend adjustments.

Estimating the SoA at the firm (-time) level can be challenging. To address this, one can arbitrarily assume either component in Eq. (2) to approximate the level of expected dividends by setting the SoA to either 0 or 1. Notably, $SoA = 0$ implies that dividend payouts follow a simple random walk process, a special case of the AR(1) process commonly assumed in the dividend growth literature. For instance, Lacerda and Santa-Clara (2010) and Yin and Nie (2021) derive expected dividend growth time-series data based on historical averages.

Overall, the straightforward structure of Lintner's payout model facilitates the forecasting of future dividends. It operates under the assumption that past dividends, expected earnings, and estimated payout ratios collectively provide sufficient information for inferring future dividend expectations. In the subsequent subsection, we present a more comprehensive model for dividend generation and explore the impact of different conditional information sets on forecast accuracy.

2.2. MSE decomposition

Assume a two-period-economy that contains the current time, denoted by t, and the future or forecasted period that is denoted by $t + h$. At the current time, there are two information sources relevant to forecast the firm i's dividend: X_i, which is public information available to everyone, and P_i, the analyst's private information that is unobservable to the public. In general, the private information can originate either from analyst's personal source of information (e.g., private communication with management (Brown et al., 2015)), or generated by her own skills used in understanding the firm's payout policy (So, 2013). Since the dividend payout for each fiscal period is fundamentally determined by the firm's management broad, the input of private information is important for analysts to generate accurate forecasts. To simplify notations, we eliminate the following time subscripts and state the first lemma:

Lemma 1 *For either linear or nonlinear information structure embedded in X_i and P_i, the true payout for firm*

i follows the true DGP without loss of generality

$$y_i = x_i + p_i + \varepsilon_i. \tag{3}$$

where:

- $x_i = \mathbb{E}(y_i|X_i)$ *is the conditional forecast based on public available* X_i.
- $p_i = \mathbb{E}(y_i|X_i, P_i) - \mathbb{E}(y_i|X_i)$ *is the separate component forecast generated by private information set* P_i.
- ε_i *is the zero mean i.i.d. error term.*

We adopt the framework proposed by de Silva and Thesmar (2021), where the three parts are assumed to be orthogonal to each other. p_i is defined differentially to capture the variations in the forecast when additional private-information is introduced. Essentially, Eq. (3) signifies that the forecast of y_i relies on information from three distinct components: publicly available data x_i (e.g., earnings and historical dividends, as previously mentioned), private information p_i exclusive to analysts, and an *i.i.d.* error term ε_i.

Expanding upon the insights from Section (2.1), investors adhering to Lintner's model can generate dividend forecasts through two distinct approaches: (a) based on past dividends, expected earnings, and the payout ratio and, (b) by relying entirely on consensus analyst forecasts. With respect to the first approach, we do not specify the method practitioners use to arrive at their expected earnings. They may employ a firm characteristic model outlined by So (2013) or directly utilize analyst earnings forecasts. However, it is worth noting that we assume the entire set of this information to be publicly available and reflected in X_i.

In the case of the second approach, which relies exclusively on the consensus analyst forecast, it is presumed to incorporate all accessible public and private information to produce the consensus dividend forecast. Nevertheless, the biases and inherent unpredictability in individual forecasts may not necessarily be fully mitigated when combined. This leads us to Lemma 2, which delineates the structure of these two forecasting methods.

Lemma 2 *Dividend forecasts that are constructed solely rely on Lintner's model* F_i^L *or the analyst consensus*

[4] *of dividend can be respectively decomposed as:*

$$F_i^L = x_i + \eta_i, \tag{4}$$

$$F_i^{AL} = x_i + p_i + b_i, \tag{5}$$

where:

- η_i *is the desperation generated by only utilizing a subset of public information in Lintner's forecast.*
- $b_i = \mathbb{E}(F_i^{AL} - y_i | X_i, P_i)$ *is the component distorted by analyst subjective bias.*

For the first Lintner's forecast, denoted as F_i^L, we introduce a random noise term η_i to account for potential distortions arising from utilizing a limited information set—namely, historical dividends and expected earnings. In the case of the analyst forecast, denoted as F_i^{AL}, we include a bias term b_i to characterize the component responsible for deviations between the analyst forecast and the true values, capturing the analyst forecast error. This structural framework implies that, despite analysts having access to both public and private information, inherent biases may cause their expectations to deviate from the true values. In practice, such biases can stem from various factors, including analysts' subjective reactions to external shocks, a tendency to maintain consistent forecast levels (as discussed by So (2013) and Hilary and Hsu (2013)), and the complexity of firms' information environments (Brown et al., 2015; Hoitash et al., 2021).

As argued by de Silva and Thesmar (2021), the framework's flexibility has a notable strength: it does not adhere to the concept of full-information rational expectations but rather accommodates deviations arising from both private information and biases. We can then formulate a forecast approach that incorporates both publicly available information and analyst forecasts following their announcements. At the end of each forecast period, two dividend forecasts become available: one is based on all historical public information X_t, while the other relies on the consensus analyst forecast. The combined forecast is derived by assigning weights to each component, as outlined in Lemma 3.

Lemma 3 *Assume the consensus forecast of dividend and all public information is available, the combined forecast (donated as* F_i^{Com}*) can be regarded to be formed by choosing a weight* β *where* $0 \leq \beta \leq 1$.

$$F_i^{Com} = (1-\beta)x_i + \beta(x_i + p_i + b_i) = x_i + \beta(p_i + b_i). \tag{6}$$

[4] To simplify, we examine the forecasting structure at an aggregate level, focusing on a representative investor and the analyst consensus, rather than delving into individual forecasts.

Under Bayesian expectations, the optimal weight is determined by the relative precision (inverse of variance) for each component of the forecast and can be calculated when the distribution of each forecast is observable (Chen and Jiang, 2006; So, 2013). However, our goal is to describe a more general framework for optimal forecasting that goes beyond the constraints of the Bayesian approach. In this broader context, we do not fix the value of the weight parameter β. Instead, our aim is to investigate how variations in this weight parameter can impact the overall Mean Squared Error (MSE) of the combined forecast. It is worth noting that when $\beta = 0$, the combined forecast corresponds to a special case of the econometric forecast as discussed in de Silva and Thesmar (2021), which relies exclusively on publicly accessible information X_i.

Based on the Eq. (4), (5), and (6), we can calculate the Mean Squared Error (MSE) for the forecasts of all the firms' future dividends using the three strategies: F^L, F^{AL}, and F^{Com}, as the evaluation of overall forecast accuracy. The cross-sectional MSE is defined as: $MSE = E[(F_i - y_i)^2]$. We assume that the chosen weight β is neither 0 nor 1, meaning that the combined forecast does not solely rely on either analyst forecasts or general forecasts based on public information.

With a large forecasting sample size, we assume that the private information component's expected value averages out across different companies, i.e., $E(p) = E[\mathbb{E}(y_i|X_i, P_i) - \mathbb{E}(y_i|X_i)] = 0$. Put it simply, we acknowledge that private information can influence dividend forecasts for individual firms, such as when analysts interact with managers to gain insights into their attitudes toward dividend payouts for the next fiscal period. However, when the sample size is fairly large, we assume that the collective effect of this private information on the forecast is distributed symmetrically. Consequently, the difference in MSE can be computed as:

$$MSE^{Com} - MSE^{L} = \beta^2 E(b_i^2) - E(\eta_i^2) - \beta(2-\beta)E(p_i^2), \tag{7}$$

$$MSE^{Com} - MSE^{AL} = (\beta-1)^2 E(p_i^2) - (1-\beta^2)E(b_i^2). \tag{8}$$

with $\beta \in (0, 1)$, Eq. (7) illustrates that the combined forecast outperforms Lintner's forecast when it is less affected by the biases in analyst forecasts, as represented by the first positive term. Additionally, it benefits more from the analyst's private information and other publicly available information that Lintner's model overlooks, as indicated by the last two negative terms. Moreover, when earnings and past dividends can only capture a small proportion of future dividends (i.e., when $E(\eta_i^2)$ is large), the combined forecast tends to generate a lower MSE. Thus, Eq. (8) suggests that the combined

forecast outperforms analyst forecasts when the advantage gained from private information cannot compensate for the increase in errors caused by bias.

Moreover, we can explore the sign of the MSE difference based on certain additional assumptions. Detailed derivations can be found in Internet Appendix Section A. We thus propose the following Lemma 4 to address this aspect:

Lemma 4 *Given the weight $\beta \in (0,1)$, assume the second moment of each component exists and is finite, the sufficient condition for $MSE^{Com} < MSE^{L}$ is:*

$$\beta < \frac{2E(p_i^2)}{E(p_i^2) + E(b_i^2)}, \tag{a}$$

and for $MSE^{Com} < MSE^{AL}$ is:

$$\beta > \frac{E(p_i^2) - E(b_i^2)}{E(p_i^2) + E(b_i^2)}. \tag{b}$$

Let's consider the two conditions under this framework. In the first condition, we can reasonably assume that for short-term horizon forecasts, such as one-quarter, two-quarters, and one-year-ahead forecasts, private information is relatively reliable, and analysts do not have significant subjective bias. Assuming that all the second-moment terms in (a) and (b) have comparable magnitudes, $E(p_i^2)$ tends to be higher than $E(b_i^2)$ in short-term forecasts. Therefore, the right-hand side of (a) is higher than 1, which naturally satisfies the first condition since β has been set to be lower than 1.

Similarly, for longer horizon forecasts (two years ahead, or beyond), private information might be sparse, and subjective bias or noise tends to be more significant. In this case, the left-hand side of (b) tends to be negative, naturally satisfying the second condition since $\beta > 0$. Therefore, for short-term forecasts, the combined forecast tends to outperform Lintner's forecast due to the advantages of private information. As the forecast horizon extends, the combined forecast naturally outperforms the analyst forecast as analysts' subjective biases increase.

Overall, we extend the MSE decomposition framework introduced by de Silva and Thesmar (2021) to demonstrate how dividend forecast accuracy benefits from combining information from both public sources and analysts. We also explain why forecasts based solely on Lintner's model or analysts may have a large MSE. Importantly, our findings do not rely on strict assumptions regarding the form of analyst beliefs or the true DGP for dividends, and they do not necessarily suggest that Lintner's or analyst forecasts underperform in all situations. Rather, our goal is to demonstrate how dividend forecasts can potentially benefit from richer information. This perspective aligns with the findings of

van Binsbergen et al. (2022), which suggest that incorporating additional features in the optimal estimator reduces errors because irrelevant information is automatically filtered out. To better capture potential nonlinear relationships and the large number of estimators contained in public information, we employ a tree-based machine learning method to construct this statistically optimal estimator. More details of our machine learning forecast formation process and the empirical results are provided in Section (3).

3. Dividend Forecast via Machine Learning

In this section, we outline our approach for constructing real-time dividend forecasts using machine-learning-based predictive regression. We employ three nonlinear and non-parametric tree-based models: Random Forest (RF), Gradient Boost Trees (GB), and Extreme Gradient Boosting Trees (XGB). Subsequently, we provide summary statistics and conduct a comparative analysis of the accuracy of our one- two- and three-quarter-ahead quarter and one- two- and three-year-ahead annual dividend forecasts against those based on Lintner's model (i.e., lagged actual dividends or expected earnings multiplied by the estimated payout ratio) and analyst forecasts of dividends. We find that our machine-learning dividend forecasts consistently outperform alternative methods.

3.1. Data for DPS forecast

We gather data from four primary sources. For the period spanning January 2002 to December 2022, we collect monthly analyst forecasts of dividend per share (DPS) data for all U.S. firms available in I/B/E/S summary file. To mitigate the influence of extreme forecasts, we use the median values as the consensus forecast. Additionally, we obtain realized DPS and earnings per share (EPS) data from the I/B/E/S actuals file. It is worth noting that we opt for I/B/E/S data instead of Compustat for actual DPS and EPS values due to differences in accounting standards used by I/B/E/S, as suggested by van Binsbergen et al. (2022). We also obtain quarterly firm fundamentals from Compustat, monthly stock prices and returns from CRSP, and monthly macro-economic indicators from the Federal Reserve Economic Data (FRED).

To optimize the utilization of available information in forecasting DPS while preserving a maximal number of observations, we focus on fundamental variables with minimal missing values that have been shown to correlate with corporate payout policy in prior studies (Allen and Michaely, 2003; Farre-Mensa et al., 2014). We also compute several financial ratios using the selected variables. A

comprehensive list of all variables used in our analysis can be found in Internet Appendix Table B.1.

For a firm to be included in our sample, it is required to have non-missing analyst forecasts of DPS, future realized DPS, and stock prices for each forecasting month. We also winsorize at the 1% and 99% level and standardize all variables, following the guidance of James et al. (2013). In cases where firm quarterly fundamentals have missing values (constituting up to 5% of the total sample), we impute these values using industry medians. Industries are classified according to the Fama-French 30 industry portfolios [5].

3.2. Forecast formation

Following the approach used by van Binsbergen et al. (2022), we employ a rolling window strategy to partition the data chronologically into training and test sets. Besides, it is also essential to ensure that the dependent variables included in the training set are accessible at the time of forecasting to prevent any look-ahead bias. To achieve this, we introduce gap years with the same duration as the forecast horizon between the training and testing periods. Consequently, each window comprises three sub-periods: the model formation year, the gap year(s), and the forecast formation year.[6] For example, suppose we are in 2004 and aim to predict one-year-ahead (2005) DPS. In this case, we would train our model using independent variables from 2002 and a dependent variable from 2003 to fit the model. We then set 2003 as the gap year, and input the independent variables of 2004 to generate the forecasts of DPS for 2005. The forecast formation for two-year and three-year-ahead forecasts follows a similar pattern.

As analysts typically provide their forecasts on a monthly basis, often before the end of each month, we ensure that our forecast is built after the announcement of the analyst forecast. For other predictors updated quarterly or annually, we use their lagged values and ensure that this information is available at each forecast time point.

To optimize our model, we employ a 5-fold cross-validation procedure and fine-tune the hyper-parameters within each rolling window estimation using the parameter ranges suggested by de Silva and Thesmar (2021).[7] The Internet Appendix Section B.3 contains detailed information about the se-

[5]The sample at certain time points may not encompass all industry categories when using more complex classification methods, such as those involving 48 or 49 industries, due to a limited number of observations. Note that the different classification methods do not generate significant effects on our forecast results

[6]We opted for a one-year training set to maximize the forecast's time horizon coverage. The overall forecast results remain stable when we experiment with two or three-year training periods.

[7]We also select optimal hyperparameters using a split-sample approach, similar to the method employed by van Binsbergen et al. (2022). The forecast results are similar.

lected hyperparameter ranges and categories for each supervised learning method. Eventually, each rolling window is constructed independently, and all forecasts are conditional and out-of-sample by design:

$$E_t^{ML}(DPS_{i,t+\tau}) = ML[X_{i,t}^{AL}, X_{i,t}^{Fin}, X_t^{Macro}],$$

where $X_{i,t}^{Fin}$ and X_t^{Macro} refer to firm fundamental and macro market variables, respectively. ML encompasses our three nonlinear supervised learning methods: random forest (RF), gradient-boost trees (GB), and extreme gradient-boost trees (XGB).

Table 1 provides a brief overview of the three sets of predictors that we employ. We utilize a total of 80 explanatory variables with varying frequencies to generate DPS forecasts across different time horizons. Our data sources include annual and quarterly firm fundamentals from CRSP/Compustat Merged database. Additionally, we manually calculate several financial ratios using the methodology outlined in the Financial Ratios Suite by WRDS. For a comprehensive list of all the variables and dataset details, please refer to Internet Appendix Section B.1.

[Table 1 about here.]

As mentioned above, we employ three ensemble tree-based regression methods, namely RF, GB, XGB to build a comprehensive forecast and assess enhancements in forecast accuracy. In essence, the ensemble regression models are constructed to mitigate overfitting by aggregating predictions from multiple independently generated trees. For the baseline model RF, each tree in the forest is built from a random sample of the data, and it makes a prediction by averaging the outputs of all the individual trees. This approach combines the simplicity of decision trees with flexibility, reducing the risk of overfitting and improving accuracy by averaging multiple deep decision trees.

By contrast, GB follows a boosting approach, where decision trees are added one at a time, and each tree attempts to correct the mistakes of the previous ones. Unlike RF, which builds trees independently, GB builds trees sequentially, with each tree designed to correct errors made by its predecessor, focusing on minimizing residuals. This iterative process continues until a predefined number of trees is reached, or a specified level of performance is achieved. XGB, an enhanced version of GB, extends the principles of gradient boosting by incorporating advanced regularization (L1 and L2), which prevents overfitting by penalizing complex models. Furthermore, XGB improves upon traditional GB by optimizing computational resources, effectively handling sparse data, and allowing for the weighting of misclassified instances, thus refining the model's accuracy and execution speed.

In summary, both GB and XGB use boosting techniques with a focus on sequential tree construction to enhance forecast accuracy. However, XGB sets itself apart by incorporating regularization and weighted sampling, making its optimization process more stable and robust. The flowchart in Figure 3 below provides a brief comparison of the working processes of the three tree-based regression models. For a more detailed explanation of the algorithms, please refer to Internet Appendix Section B.2.

[Figure 3 about here.]

3.3. Forecast results and evaluation

We train the machine learning regression model starting in January 2002, following the formation process mentioned above. Eventually, we collect a dataset containing 534,255 monthly one-year-ahead DPS forecasts for 7,213 unique firms and 316,083 monthly three-year-ahead forecasts for 5,047 unique firms.

Figure 3 illustrates an example of a regression decision tree used to extract one-year-ahead DPS forecasts from the RF Regression model. To simplify the structure, we limited the tree's depth to 3 levels. Starting from the root node, the initial selected feature is the analyst forecast of DPS, and the splitting is based on a threshold value of 1.0. The data then proceeds to subsequent nodes, each representing a further branch of the feature space, guided by additional feature thresholds. This sequential partitioning continues until a terminal leaf node is reached.

[Figure 4 about here.]

Table 2 presents a summary of statistics and accuracy evaluations for all the forecasts. In the top panel, we observe that the analyst forecast (AF) closely approximates the actual value ($Actual$) for short-term forecasts (quarters ahead) but tends to deviate as the forecast horizon extends to 3 years. The forecasts generated based on the dividend target on earnings using the long-term payout ratio are denoted as $EPS * payout$. To calculate a valid payout ratio, we replace the negative earnings with 0.06 $\times$ total assets as Hou et al. (2012) and Guay et al. (2011) do. The payout ratio is estimated as the sum of past DPS and repurchases divided by EPS, with this ratio constrained to fall within the range of 0 to 1. Interestingly, forecasts produced by this approach consistently diverge the most from the actual values across various forecasting horizons.

Compared to these two commonly used approaches, dividend forecasts generated using three machine learning methods, namely RF, GB, and XGB, exhibit more consistent mean values that closely match the actual values and are less influenced by the forecast horizon's length.

[Table 2 about here.]

We assess the overall quality of our forecasts by considering two key metrics: Mean Squared Error (MSE) and the percentage of improvement compared to a naive forecast based on the assumption of complete dividend smoothing, where future DPS is assumed to be equal to the lagged period of realized DPS (Yagan, 2015). Based on these two metrics, forecasts derived from earnings multiplied by an estimated payout ratio yield the highest MSE and fail to surpass the benchmark based on lagged DPS at any forecast horizon. This result sheds light on two limitations within the existing literature. First, it suggests that actual dividend payouts by firms may not consistently follow historical payout ratios, which aligns with the reduced significance of the target payout ratio, as indicated by survey results (Brav et al., 2005). Second, we employ the common practice of substituting negative EPS and abnormal DPS (e.g., for firms with a history of negative EPS but maintaining a high level of dividend payment) in line with existing literature (Li et al., 2013; Bilinski and Bradshaw, 2015). However, our results indicate that this approach may further introduce errors.

By contrast, analyst forecasts exhibit a lower MSE when compared to forecasts derived from earnings. However, it is essential to note that analyst forecasts demonstrate increasing MSE as the forecast horizon extends. For instance, the MSE increases from 0.008 for one quarter ahead to 0.400 for three years ahead. More importantly, at the three-year-ahead forecast horizon, analyst forecasts fail to surpass the performance of the naive benchmark, resulting in a 39.2% higher MSE compared to the benchmark.

Our machine learning forecasts (RF, GB, and XGB) consistently yield the lowest MSE values, outperforming both the lagged DPS benchmark and the analyst forecasts across all forecasting horizons. As analyst forecasts serve as critical input predictors in generating ML forecasts, the accuracy of ML forecasts also tends to decrease as the forecasting horizon increases. However, because our input predictor sets include other essential variables such as EPS, lagged DPS, and stock price, our forecasts can maintain a robust and superior performance compared to those relying on limited information, such as analyst forecasts.

Among the three machine learning methods, XGB appears to perform the least effectively, producing a higher MSE of 0.113 at the one-year-ahead forecast horizon, even exceeding that of the analyst

forecast. Hence, we create a composite machine learning forecast, denoted as $Comp$, by averaging the predictions from RF and GB. Our findings indicate that this composite approach consistently delivers relatively more accurate forecasts compared to relying solely on either of the two individual machine learning methods. The detailed performance comparison of our composite machine learning forecast and the analyst forecast at different forecasting horizons is also illustrated in Figure 5.

[Figure 5 about here.]

Overall, our findings indicate that machine learning forecasts, which incorporate analyst forecasts alongside other publicly available information, consistently outperform all other forecasting methods across a range of forecasting horizons. While analyst forecasts demonstrate strong performance in the short term, their informational value diminishes as the forecast horizon extends. Additionally, the product of earnings forecasts and payout ratios does not prove to be an effective method for dividend forecasting.

Having established the superior performance of machine learning forecasts over alternative methods, it is crucial to examine the key contributors to the construction of dividend forecasts. Figure 6 presents the feature importance analysis for one and two-year-ahead DPS using the RF forecast method. [8]

[Figure 6 about here.]

Impurity importance is a measure defined by the extent of the decrease in the sum of squared errors when a specific feature is used to split within each tree individually and then averaged across the entire forecasting model. The analyst forecast of dividends, as well as the realized dividends from the past year and quarter, emerge as the most critical features, collectively accounting for over 90% of the impurity importance. Interestingly, for the two-year-ahead forecast, the significance of past DPS surpasses that of the analyst forecast. Additionally, analyst forecasts of earnings and realized earnings are found to contain valuable information for predicting future dividends. Other fundamental variables related to the firm, such as stock price, operational EPS, debt levels, and tax payments, also contribute significantly to the forecasting outcomes. These results reinforce the rationale of using realized dividends or expected earnings as indicators for inferring future dividend payouts. However, they also emphasize the importance of incorporating a broader range of features to enhance the accuracy and comprehensiveness of dividend forecasts.

[8] The feature importance results obtained from GB and XGB exhibit similar patterns.

In addition, we employ Partial Dependence Plots (PDPs) to provide further insights into the marginal contributions of key features to the output forecast. PDPs help us gain a deeper understanding of the relationship between the response variable and specific features of interest while taking into account the average effect across the distribution of other variables in the dataset.

We select four key variables based on the previously determined feature importance: analyst forecasts of DPS and EPS, past realized DPS, and stock price. As illustrated in Figure 7, the PDPs reveal significant nonlinearity in the relationship between future DPS and these four key features. This empirical evidence, highlighting non-linear patterns in the relationships between features and the outcome variable, may underpin one of the primary reasons for the enhanced forecasting accuracy achieved through the utilization of machine learning and provides further support for the application of nonparametric tree-based models in forecasting dividends.

[Figure 7 about here.]

4. Why Does Machine Learning Outperform Traditional Methods? Evidence from Firm Complexity.

So far, our results indicate that machine learning-based dividend forecasts outperform traditional approaches, such as analyst forecasts. In the theoretical framework described in Section 2, we propose an economic mechanism to explain this outcome. The core concept is that machine learning models have the capacity to leverage more comprehensive information for forecasting compared to humans. In this section, we empirically test this hypothesis.

Prior research demonstrates that analysts often incur larger EPS forecast errors when they produce forecasts for more complex firms, such as conglomerates or multi-segment companies, possibly due to a lack of industry-specific expertise (Barinov et al., 2022). Studies by Lehavy et al. (2011) and Loughran and McDonald (2020) also indicate that various measures of analyst forecast imprecision are inversely related to firm complexity. This discussion implies that firm complexity serves as an ideal setting to test our main hypothesis regarding the superior ability of machine learning to process complex information. To be precise, if machine learning does indeed possess the capacity to effectively utilize more comprehensive information compared to humans, then the accuracy of machine-generated forecasts should be less affected by the complexity of firms' information environments when compared to forecasts produced by humans. This is the main hypothesis that we examine in this section.

Firm complexity is commonly measured using proxies such as firm size and the number of segments. However, Hoitash and Hoitash (2018) introduce a novel metric called account reported complexity (ARC), which is based on the count of monetary items disclosed in eXtensible Business Reporting Language (XBRL) within company filings. ARC is derived from disclosures that encompass a wide array of items, including investments, R&D expenses, and acquisitions and it effectively captures the overall complexity of a firm's structure. We employ this measure in our study because it more directly captures economically significant information disclosed in company filings.

We collect annual ARC data from the Hoitash & Hoitash website.[9] The ARC dataset covers the period starting from 2009 and includes multiple measures derived from public data available on the SEC's website. For our analysis, we take the main measure *ARC*, to proxy firm complexity. Subsequently, we calculate annual squared forecast errors for three-year-ahead DPS forecasts for each firm in our sample by averaging the available monthly squared forecast errors. We choose to assess the accuracy of three-year-ahead DPS forecasts because our earlier findings suggest that analysts face challenges in providing accurate forecasts at longer horizons. This may be attributed to the requirement of analyzing a more extensive dataset when forecasting dividends over longer time horizons; indeed, this is the concept we aim to test. To link our sample with the ARC dataset, we use firms' fiscal year and CIK code, which the ARC data uses as identifiers. This matching process results in a dataset comprising 24,234 firm-year observations spanning from 2009 to 2021.

We conduct two tests. In the first univariate test, we sort the firms by ARC and split them into five portfolios, indicating rising complexity from the first to the fifth group. As shown in Table 3, Panel A, we observe a clear, monotonically increasing trend in analyst forecast errors along with increasing firm complexity. Notably, there is a significant difference of 0.088 in the time-series Mean Squared Error (MSE) for analyst forecasts between the most complex and simplest groups, as indicated in Column C-S of the table. This observation aligns with our expectation, suggesting that analysts face challenges in producing dividend forecasts for firms with more complex information environments.

We do not observe the same pattern when we examine the machine learning forecasts. First, we find that machine learning forecast errors consistently remain lower for each complexity group and exhibit smoother changes across different groups. Furthermore, for machine learning forecasts, the difference in MSE between the most complex and simplest firms is not statistically significant and it is also about 63% lower than the difference observed for analyst forecasts.

As a more comprehensive test, we construct annual cross-sectional regressions of the two squared

[9] https://www.xbrlresearch.com/firm-complexity/

errors on ARC for each year, similar to the approach used by Fama and MacBeth (1973) in the Fama-MacBeth regression. In Table 3, Panel B, Column (A) presents the regression results on ARC without control variables, while Column (B) includes controls. In addition to firm fundamentals, we incorporate two dummy variables, *Conglo* and *GeoMulti*. Following Barinov et al. (2022), *Conglo* takes a value of one if the firm is a conglomerate with multiple business segments in the same year, and *GeoMulti* is one if the firm generates sales from multiple geographic segments.

Our findings reveal that, regardless of whether controls are included or not, machine learning forecast errors do not exhibit a significant slope with respect to ARC or other complexity measures. In contrast, analyst forecast errors show a significant positive association with firm complexity, and this relationship remains consistent after controlling for other variables such as firm size and book-to-market ratio. The estimated slope suggests that, holding other variables constant, a hundred-unit increase in monetary items disclosed in XBRL results in a 0.035 increase in analyst forecast squared errors of DPS. As for the other two dummy measures, neither of them exhibits significance in forecasting errors.

[Table 3 about here.]

In summary, the findings in this section suggest that the advantages of employing machine learning for dividend forecasts stem from the machine learning algorithms' ability to effectively analyze more comprehensive information. Therefore, the benefit of using machine learning in dividend forecasting is particularly pronounced for firms with more complex information environments.

5. Application: Estimating the Expected Return

Our results thus far indicate that machine learning forecasts of dividends exhibit lower forecast errors, primarily attributed to machines' superior ability to process more comprehensive information compared to humans. This finding holds significant potential for various economic applications. In this section, we focus on one such application: estimating expected returns.

Expected return is a fundamental concept in finance and a crucial input in investment decision-making. However, estimating expected returns is a challenging task. One of the main difficulties arises from the forward-looking nature of expected returns, while many finance models, such as the Capital Asset Pricing Model (CAPM), heavily rely on historical data. To address this challenge, a strand of literature proposes an approach to estimate forward-looking expected returns based on forecasts, commonly referred to as the implied cost of capital (ICC) in the literature. This approach

has been shown to measure expected returns effectively (Lee et al., 2009; Hou et al., 2012; Li et al., 2013).

Dividend forecasts are a crucial input in forecast-based expected return models. Given that our machine learning models can generate dividend forecasts with reduced errors, this may also allow us to obtain a more accurate expected return. Therefore, in this section, we incorporate our machine learning-generated dividend forecasts into the computation of forecast-based expected returns. First, we revisit the rationale behind forecast-based expected returns as a suitable proxy for expected returns. We then compare our expected return measure against the traditional method of computing ICC, which relies on forecasts of EPS and the payout ratio. Our findings demonstrate that expected returns calculated using our DPS forecasts outperform the alternatives as a proxy for expected returns.

5.1. Expected Return Imputed from Free Cash Flow Discount Models

Conceptually, the implied cost of capital (ICC) is defined as the value of the internal return, denoted as r_{ICC}, that solves the infinite Gordon dividend discount model:

$$P_t = \sum_{k=1}^{\infty} \frac{E_t[DPS_{t+k}]}{(1+r_{ICC})^k}, \tag{9}$$

which indicates that the present value of equity price P_t equals the sum of all the discounted future expected dividend payouts $E_t[DPS_{t+k}]$. Combining this with the log-linear approximation demonstrated in Campbell and Shiller (1988), Pástor et al. (2008) shows that under the assumption that both dividend growth and expected returns follow a stationary AR(1) process, r_{ICC} is perfectly correlated with the expected return μ_t:

$$r_{ICC} = \frac{\kappa}{1-\lambda} + (\mu_t - \frac{\kappa}{1-\lambda})\frac{1-\rho}{1-\rho\lambda}, \tag{10}$$

where κ and λ are constants determined by the AR(1) process of μ, and $\rho = 1/(1+exp(\bar{d}-p))$. Therefore, ICC is considered a natural proxy for expected return, as it encapsulates information about future returns without relying on the noisy realized returns (Gebhardt et al., 2001; Easton and Monahan, 2005).

In practice, r_{ICC} can be empirically determined by solving the transformed finite Gordon model, which decomposes the original Eq. (9) into a sum of discounted expected free cash flow terms

$EFCF_{t+k}$ (i.e., firm payouts) and a discounted terminal value term TV_{t+T}.

$$P_t = \sum_{k=1}^{T} \frac{E_t[EFCF_{t+k}]}{(1+r_{ICC})^k} + \frac{TV_{t+T}}{(1+r_{ICC})^T}. \tag{11}$$

We initially employ the finite Gordon model to calculate two ICCs (denoted as r_e) as the benchmark expected returns for comparison with our forecasted dividend-based expected return. The first ICC is generated using the analyst forecasts of earnings times the payout ratio as $EFCF$ (Pástor et al., 2008; Lee et al., 2009; Li et al., 2013). This computation is based on three main assumptions related to the growth process of earnings and the payout ratio, as well as the imputation of abnormal values, as discussed in Table 4. For the calculation of the second ICC, we replace the analyst forecasts of EPS with random forest forecasts using the methods outlined in van Binsbergen et al. (2022). The substitution of $EFCF$ is driven by the aim of mitigating the subjective bias that may arise from analyst forecasts of earnings, which has been identified as a key factor distorting the effectiveness of ICC as a proxy for expected return (Lee et al., 2020). We follow the approach introduced in Pástor et al. (2008) to obtain the two baselines r_e, and the detailed computation process is contained in Internet Appendix Section C.

For the calculation of our dividend-based ICC (denoted as r_d), we utilize the composite machine learning forecasts of one, two, and three-year-ahead dividends as a proxy for $EFCF$. In this case, we assume that dividend grows perpetually at the average GDP growth rate. Therefore, the computation of our ICC relies solely on available information without the need for imputation of long-term cash flow expectations.

[Table 4 about here.]

The calculation details of r_d are provided in Table 4. We first require the input of one-year-ahead and two-year-ahead machine learning forecasts of DPS to be non-missing and non-zero. To address the issue of missing and zero forecasts, which represent approximately 7% of the total sample, we employ an imputation technique. Specifically, we replace these missing or zero forecasts with a value equivalent to four times the one and two-quarter-ahead forecasts, assuming that these firms will make annual dividend payments consisting of four identical quarterly dividends. For the firms without effective forecasts of three-year-ahead DPS, which comprises 14.5% of the total sample, we substitute them with the the two-year-ahead DPS forecasts times the last gross growth rate $((E_t[DPS_{t+2}])^2/E_t[DPS_{t+1}])$, assuming they maintain the same growth rate from the second to the

third year. This approach is in line with Pástor et al. (2008). We employ the 30-year rolling average of the real GDP growth rate as a proxy for the perpetual growth rate of DPS.

We then compute r_d using the constant dividend growth model with the three-year DPS forecasts. The resulting ICC values are constrained within the range of 0 to 0.3, following Hommel et al. (2023), and approximately 3.9% of observations are removed due to this trimming process.[10] This restriction is applied to ensure that the ICC values are meaningful and do not include negative values, as a negative ICC is not interpretable as a discount rate, and excessively high ICC values may indicate unreliable high reinvestment rates. In total, we obtain 446,696 DPS-based ICC values for 6,426 unique companies spanning from January 2005 to December 2021.

5.2. Testing Our Expected Return: In-Sample Analysis

In the first test, we investigate the predictability of future realized returns in the aggregate market, as represented by the S&P 500 index, using the aggregate expected return measures. To construct the aggregate expected return, we consider all available constituents of the S&P 500 index for each month spanning from January 2005 to December 2021. We employ both equally-weighted and value-weighted aggregation approaches. After aligning our sample with the S&P 500 companies listed in the CRSP index constitution data file, we find that, on average, our sample comprises 469 firms each month. This sample coverage represents approximately 92% of the total market capitalization.

Prior to examining the relationship between the aggregate expected return and future realized returns, we present a time-series of the S&P 500 value-weighted expected return derived from machine learning forecasts of DPS in Figure 8, from January 2005 to December 2021. Of particular note is the peak of our expected return, which occurs at 7.4% in April 2009, coinciding with the near end of the 2008 financial crisis. During the two NBER recession periods that fall within our study's time frame, the expected return measure demonstrates a significant increase, consistent with the counter-cyclical behaviour of expected returns as suggested by existing studies (e.g., Li et al. (2012), Li et al. (2013)). Following the first recession, the expected return gradually decreases until the outbreak of the COVID-19 pandemic in February 2020.

[Figure 8 about here.]

Next, we run the following univariate predictive regression framework for different expected re-

[10] Other methods of handling extreme values, such as winsorization at the 1% level, do not significantly affect the subsequent application results.

turn proxies as suggested by Fama and French (1988):

$$\sum_{h=1}^{H} \frac{r_{t+h}}{H} = \beta_0 + \beta_1 \times M_t + \epsilon_{t+h}, \tag{12}$$

where r_{t+h} represents the continuously compounded monthly realized returns, either equal-weighted or value-weighted, in excess of the compounded three-month T-bill rate. We present the results for three different monthly prediction horizons, namely $H = 1, 6, 12$, in the interest of brevity, although the results hold at longer horizons as well.

M_t donates the vector of expected return proxies. Our primary expected return proxy relies on machine learning-generated dividend forecasts, and we assess its performance in comparison to expected returns computed using earnings forecasts and various widely-used valuation ratios, consistent with the methodology detailed in Welch and Goyal (2008). Specifically, we include dividend-price ratio (dp), earnings-price ratio (ep), payout ratio (py), default spread ($Default$), and term spread ($Term$). We compute the aggregate market ratios dp, ep, and py using the aggregate dividend and earnings data obtained from Michael R. Roberts' website.[11] $Default$ is determined as the difference between BAA and AAA-rated corporate bond yields, while the $Term$ is calculated as the difference between the 10-year Treasury yield and the three-month bill rate. We source all macroeconomic data from the Federal Reserve Economic Data (FRED) database.

Table 5 presents summary statistics for the relevant variables. The two benchmark expected return measures are calculated using an EPS-based model, as listed in Table 4. These calculations involve machine learning techniques and analyst forecasts of EPS. Overall, the DPS-based expected return has a lower mean and standard deviation compared to both the equally-weighted and value-weighted EPS-based expected returns. Specifically, the DPS-based equally-weighted expected return has a mean and standard deviation of approximately 5.2% and 0.003, respectively, which is relatively lower than the EPS-based equally-weighted expected return, with a mean and standard deviation of 8.8% and 0.014.

All expected return measures exhibit persistence, with a first-order autocorrelation exceeding 0.92, consistent with the expected characteristics of a robust expected return proxy, as suggested by Lee et al. (2010). This autocorrelation remains significant even at 12 lags. The ADF statistics and corresponding p-values indicate that we can reject the null hypothesis of a unit root for r_d, $Default$, D/P, and $payout$ at least at 10% level. However, we fail to reject the null hypothesis for all the EPS-based

[11] http://www.econ.yale.edu/~shiller/data.htm

expected return measures, *Term*, and the E/P ratio.

[Table 5 about here.]

Table 6 tabulates the results of univariate predictive regressions using Eq.(12) to forecast aggregate realized market returns. The left panel presents the outcomes of regressing equally-weighted returns ($ewret_{t+h}$) on predictors, while the right panel uses value-weighted returns ($vwret_{t+h}$) as the dependent variable. We report the Newey and West (1987) t-statistic with lag numbers equal to the forecasting horizon length.

In general, our findings indicate that short-term return predictions are relatively modest. However, even in the short term, the machine learning-forecasted DPS-based expected return (r_d) consistently outperforms other proxies. Specifically, for the one-month-ahead forecasting of $ewret_{t+1}$ ($vwret_{t+1}$), r_d demonstrates a weakly significant coefficient of 3.145 (2.191), with corresponding t-statistics of 1.822 (1.654) and the highest R-squared of 4.34% (2.66%). In contrast, the EPS-based expected return measures ($r_{e,AL}$ and $r_{e,ML}$) do not predict one-month-ahead returns effectively. Similarly, the three valuation ratios and two business cycle indicators also fail to forecast one-month returns.

For longer horizons, such as 6 and 12 months ahead, we generally observe higher R-squared and t-statistics for all predictors. However, at these extended forecast horizons, r_d continues to outperform other expected return measures and other predictors. They consistently exhibit the highest t-statistics and R-squared values. This outcome underscores the clear advantage of using expected returns calculated through machine learning-generated dividend forecasts over other benchmark expected return measures in predicting future realized returns.

[Table 6 about here.]

It is also crucial to assess whether r_d retains predictive power for future returns when considered alongside other predictors. To examine this, we conduct bivariate regressions of realized returns on r_d in conjunction with another predictor. Specifically, we perform the following bivariate predictive regression:[12]

$$r_{t+1} = \beta_0 + \beta_1 \times r_{d,t} + \beta_2 \times N_t + \epsilon_{t+h}, \tag{13}$$

where N_t takes values from the set $r_{e,AL}, r_{e,ML}, D/P, Term$. As Table 7 illustrates, the coefficient of r_d consistently maintains its statistical significance, with the lowest recorded t-statistic being 1.78 when

[12]For the sake of brevity, we present results for one-month-ahead forecasts, as results for longer horizons align with our conclusions.

another predictor is included in the regression. Interestingly, when we include $r_{e,AL}$ alongside r_d, it becomes statistically significant at the 10% level. However, the sign also changes to negative, indicating that future realized returns exhibit a negative covariance with the expected return calculated by using earnings forecasts, contrary to the theoretical assumption discussed in Section 4.1. These changes may be attributed to the high correlation between r_d and r_e. In summary, the results in this section reinforce the argument that r_d possesses relatively robust predictive power for future returns compared to its competitors.

[Table 7 about here.]

5.3. Testing Our Expected Return: Out-of-Sample Analysis

Having established the superiority of our machine learning forecasted DPS-based expected return through in-sample regression and time-series measurement error variance analyses, the next step is to assess its performance in out-of-sample predictions. This evaluation focuses on the model's ability to make predictions based solely on historical information available at each forecasting time point. This test holds significance because previous studies, such as Welch and Goyal (2008), show that while expected return predictors work well in in-sample, they tend to perform poorly in out-of-sample. In this test, we use a forecast period from January 2010 to December 2021, with a 60-month estimation window, as well as an forecast period from January 2015 to December 2021, employing a 120-month estimation window.

The out-of-sample R^2 using a length of m estimation window is expressed as (Campbell and Thompson, 2008):

$$R^2_{OS} = 1 - \frac{\sum_{k=1}^{T-m}(r_{m+k} - \hat{r}_{m+k})^2}{\sum_{k=1}^{T-m}(r_{m+k} - \bar{r}_{m+k})^2}. \tag{14}$$

We employ both rolling and expanding window methods to fit the model and predict future returns, a technique also employed by Huang and Kilic (2019). In the rolling window method, the first out-of-sample forecast $\hat{r}_{m+1}$ is obtained using parameters estimated from estimation using the initial m observations. Subsequently, the second forecast is obtained with the estimation using the next m observations start from the second time point, and so on. Concurrently, the historical average $\bar{r}_{m+k}$ is computed as the average excess return over the most recent m periods. In the expanding window method, we modify the approach by using all available past information for estimating both $\hat{r}_{m+k}$ and $\bar{r}_{m+k}$. As an evaluation criterion, a positive R^2_{OS} indicates that the predictor results in a lower Mean

Squared Error (MSE) compared to predictions based on a simple historical average.

We also conduct a test to examine the null hypothesis, where $R^2_{OS} < 0$, as opposed to the alternative hypothesis, where $R^2_{OS} > 0$. This test employs the adjusted Mean Squared Prediction Error (MSPE) statistic introduced by Clark and West (2007):

$$f_{t+1} = (r_{t+1} - \bar{r}_{t+1})^2 - [(r_{t+1} - \hat{r}_{t+1})^2 - (\bar{r}_{t+1} - \hat{r}_{t+1})^2]. \tag{15}$$

f_{t+1} is subsequently regressed on a constant term, and the test relies on the p-value obtained from a one-sided t-statistic.

[Table 8 about here.]

Table 8 presents the R^2_{OS} values and their corresponding p-values for a one-sided test applied to one-month-ahead equal and value-weighted returns. When employing a 60-month rolling window and 120-month rolling window for out-of-sample forecasts of equal-weighted returns, r_d delivers the highest positive R^2_{OS} values, reaching 5.54% and 6.41%, respectively. Importantly, all of these R^2_{OS} values are statistically significant, with a minimum significance level of 5%, as determined by the adjusted-MSPE statistic's p-values. In the case of forecasting value-weighted returns, r_d continues to produce positive and statistically significant R^2_{OS} values when using a rolling window approach. Transitioning to the expanding window approach, r_d remains positive but does not attain statistical significance.

On the other hand, both r_e variables fail to produce positive and statistically significant R^2_{OS} values in all instances. Interestingly, in out-of-sample forecasting, $r_{e,AL}$ generally outperforms $r_{e,ML}$, suggesting that a more accurate measure of EPS expectations may not necessarily enhance the ability to capture the out-of-sample dynamics in forecasting realized returns. Among the other predictors, it is observed that $Default$, D/P, and $payout$ occasionally produce positive and significant R_{OS} values. However, none of them consistently achieve positive and statistically significant R^2_{OS} values across all cases.

One limitation of the out-of-sample tests, as highlighted by Welch and Goyal (2008) and Huang and Kilic (2019), is its sensitivity to the choice of estimation method and forecast period, a point that our results also confirm. To address this limitation and ensure the consistency of superior out-of-sample performance, we examine the differences between the cumulative squared forecast errors obtained using the historical average return and those obtained using the tested expected return proxy.

In this test, a positive slope indicates a sustained advantage in out-of-sample forecasting compared to the historical average.

Figure 9 illustrates the findings. Our expected return measure maintains a consistently positive slope in the forecast period spanning from January 2010 to December 2021. In contrast, the expected return imputed from analyst forecasts of earnings displays a negative difference in cumulative squared error from 2013 to 2020, while the expected return imputed from machine learning forecasts of earnings exhibits an increasing error in recent years. Among the other expected return proxies, only D/P surpasses the historical average return during the tested period, but the difference remains consistently close to zero. Consequently, we confirm that the machine learning-forecasted DPS-based expected return demonstrates relatively robust out-of-sample return predictability.

[Figure 9 about here.]

5.4. Testing Our Expected Return: Variance of Measurement Errors

In the preceding section, we demonstrate the superiority of our DPS-based expected return by showcasing its strong track record and superior predictability of future realized returns via regression tests. In this section, we further evaluate its performance using a novel metric, namely the variance of measurement errors. Lee et al. (2020) introduce a parsimonious framework based on treatment effect studies, suggesting that minimizing measurement error variance (MEV) is necessary and sufficient for unbiased expected return proxies. The authors argue that relying solely on general evaluations derived from predictive regression, such as mean squared error and the significance of slope coefficients, is inadequate and can potentially lead to misleading conclusions. While our paper does not specifically assign a treatment effect, we employ this MEV-based evaluation as a valuable complementary assessment for our expected return proxy, derived from machine learning-based dividend forecasts.

Lee et al. (2020) define that if there is lack of correlation between the news of treatment effect and expected return, the time-series MEV for firm i takes the form of Equation (16):

$$MEV_i = var_i(\hat{er}_{i,t}) - 2cov_i(r_{i,t+1}, \hat{er}_{i,t}) + var_i(er_{i,t}), \tag{16}$$

where $var_i(\hat{er}_{i,t})$ donates the time-series variance of the tested expected return measure, $cov_i(r_{i,t+1}, \hat{er}_{i,t})$ is the time-series covariance between the tested expected return measure and future realized return,

$var_i(er_{i,t})$ is the variance of the true unobservable expected return. Expected return proxy with lower MEV is considered more informative about time-series variations in expected return and leads to less biased estimates of treatment effects.[13]

The rationale for constructing MEV_i is straightforward. It posits that the variance of the measurement error of any expected return proxy rises with the proxy's volatility while declining with the correlation between the expected return proxy and future realized returns. When the expected return proxy closely aligns with future realized returns, the time-series variation in the expected return proxy is more likely to capture the fluctuations in the firm's true expected return. Offering an alternative mathematical perspective, the construction of MEV_i implies that if the volatility of the expected return proxy fails to translate into a higher covariance between the expected return proxy and the true future return (i.e., when the correlation is low or even negative), the measurement error for this proxy is higher.

The third term, $var_i(er_{i,t})$, remains constant across different expected return measures. Therefore, our focus can be directed towards comparing the first two terms of different expected return measures and utilizing the scaled measurement error variance for assessment:

$$SMEV_i = var_i(\hat{er}_{i,t}) - 2cov_i(r_{i,t+1}, \hat{er}_{i,t}) = \sigma_i^2(\hat{er}_{i,t}) - 2corr_i(r_{i,t+1}, \hat{er}_{i,t})\sigma_i(\hat{er}_{i,t})\sigma_i(\hat{r}_{i,t}). \tag{17}$$

[Table 9 about here.]

Table 9 presents the time-series variance of different expected return proxies, denoted as $Var(\hat{er}_t)$ [14], along with the covariance between expected return proxy and realized returns at various future horizons, $Cov(r_{t+h}, \hat{er}_t)$. Additionally, it displays the calculated scaled measurement error variance, $SMEV_i$, using Eq. (17). It is worth noting that for a constant expected return, equivalent to predicting future returns based on a mean-reversion process, the SMEV will be zero. As a result, an expected return measure with a more negative SMEV is considered a superior expected return proxy.

In the left panel, when using the tested variable to proxy the equal-weighted return, we observe that r_d and D/P exhibit negative SMEV values across all horizons. Particularly, for one-month-ahead *ewret*, r_d displays the most negative SMEV value, amounting to -0.0059. In contrast, both EPS-based expected return measures exhibit positive SMEV values across all horizons, indicating their poor

[13] For a detailed derivation establishing the link between MEV and treatment effect, see Lee et al. (2020).

[14] Note that the variances of expected return proxies at different horizons are not exactly the same due to decreasing sample sizes with longer horizons. However, the magnitudes of the variances do not vary significantly.

performance of these measures in the time-series dimension. When we use value-weighted returns, r_d again consistently exhibits the most negative SMEV. Among other predictors, only D/P manages to attain a negative SMEV, particularly when proxying returns at 6 and 12 months ahead.[15]

In summary, the expected return proxy calculated from machine learning-based dividend forecasts stands out as the top-performing expected return measure, exhibiting the lowest time-series SMEV. This result remains robust across various horizons and applies to both equal-weighted and value-weighted returns. On the other hand, several other expected return proxies, including r_e, E/P, and *payout*, tend to display higher variances. However, this high variance does not translate into a stronger covariance with realized future returns. This result can be attributed to various factors. For instance, in the case of r_e, the volatility may stem from the noise in analyst forecasts of earnings and the challenges associated with accurately estimating the plowback rate (payout ratio). Our machine learning-based DPS forecast offers an advantage by directly bypassing the need to estimate the plowback rate, thereby reducing errors in the expected return proxy.

5.5. Testing Our Expected Return: Portfolio Analysis

Thus far, we have established the superiority of our expected return proxy derived from machine learning-based DPS forecasts as a time-series expected return proxy. However, all these results are at the aggregate market level. Existing literature also underscores the importance of conducting similar tests at the portfolio level (Li et al., 2013; Easton and Monahan, 2005; Guay et al., 2011; Lee et al., 2020). Therefore, it is crucial to demonstrate that the superior performance of our expected return proxy persists when we conduct our analysis at a portfolio level.

In this test, we initially rank all the stocks traded on the NYSE, NYSE American, and Nasdaq based on their size and book-to-market ratio each June from 2005 to 2021 in line with Li et al. (2013). We then create three size categories (small, medium, and large) and three value categories (low, medium, and high) using the NYSE breakpoints for market capitalization and book-to-market ratio. Specifically, a stock is assigned to the small size portfolio (S) if its firm capitalization is equal to or lower than the bottom 30% breakpoint, to the medium size portfolio (M) if it falls within the middle 40%, and to the large size portfolio (L) if it falls within the top 30%. The assignment for the high (H), medium (M), and low (L) value portfolios follows the same methodology.

Next, we align these categorized samples with our DPS and EPS forecast samples, ensuring that

[15] *payout* displays a much larger variance and SMEV compared to the other predictors. This discrepancy arises from the substantially higher magnitude of *payout* originally, as evident in the summary statistics presented in Table 5.

we include only firms with valid expected return estimations. This process results in an average of approximately 2,100 firms for each size or value portfolio when testing the DPS-based expected return and around 3,000 firms for each portfolio when testing the two EPS-based expected return measures. To conduct the predictive regression test, we calculate the portfolio expected returns and excess returns by applying value-weighting to the individual firm-level data.

[Table 10 about here.]

The top panel of Table 10 presents regression results for portfolios categorized by value, while the bottom panel focuses on portfolios categorized by size. In most cases, we observe that r_d exhibits statistically significant predictive power across all six portfolios and forecasting horizons, with the exception of the one-month ahead returns for the small size, medium value, and high-value portfolios. Both r_e measures also exhibit overall good predictive performance, but their performances are less effective in forecasting short-horizon returns for large portfolios and low-value portfolios. To provide a comprehensive evaluation of their performance, we calculate the average estimated coefficient β_1 and R^2 for different horizon forecasts within each group. These calculations reveal that when r_d is used as the predictor, it achieves much higher average R^2 values, with the exception of the small portfolios. These results underscore the superior predictive capabilities of r_d compared to r_e across various portfolio groups and forecasting horizons.

5.6. "Horse race" between various expected return proxies

In this subsection, we summarize the key findings from all the tests conducted to compare our expected return proxy calculated using machine learning-based DPS forecasts against other benchmark expected return measures. For the aggregate market return represented by the S&P 500 index, the evaluation metrics are three-fold: in-sample predictive power, in-sample variance of measurement error under the treatment effect framework, and out-of-sample predictive power. We show that our expected return proxy exhibits strong in-sample predictive power for both equal and value-weighted aggregate returns across various forecasting horizons, outperforming other tested proxies. Bivariate regression tests and the in-sample measurement error variance analysis support this result.

Furthermore, the out-of-sample forecasting results consistently demonstrate the overall superiority of our expected return measure. This superiority is evident through positive out-of-sample R-squares and significant MSPE statistics, and it holds across different estimation windows. Additionally, we expand our analysis to assess the predictive power of our expected return measure for

portfolio returns, categorized by market cap and book-to-market ratio. We find that it maintains strong predictive performance in these portfolio-based tests as well. Lastly, we also test the predictability of the aggregate market return conducted by the CRSP index using the aggregate expected return measures constructed from the entire sample. This approach enables us to assess the robust performance of our expected return proxy without being constrained by the S&P 500 universe. We demonstrate that our expected return measure remains the top performer in both in-sample and out-of-sample tests, as shown in Internet Appendix Section D.

These results suggest that using machine learning forecasts of dividends as an input in forward-looking expected return measure estimations is a useful approach that can potentially yield more accurate expected returns. We believe this approach addresses two limitations of the traditional earnings-based methods. First, in the earnings-based methods, calculating dividends from earnings forecasts involves multiplying earnings forecasts by the payout ratio. However, payout ratios are often estimated based on past realized values, making them less forward-looking and potentially introducing bias and noise. Additionally, some studies (e.g., (Pástor et al., 2008; Li et al., 2013; Hou et al., 2012)) replace negative earnings forecasts and select constant payout ratios arbitrarily, further introducing bias. In contrast, our machine learning-based dividend forecast, which leverages a wider range of information, avoids these issues and reduces noise in the data.

6. Conclusion

Accurate dividend forecasts are crucial for informed decision-making in capital allocation and risk management, as well as for asset valuation. The existing literature employs three common dividend forecasting measures: (i) past dividends, (ii) the products of earnings forecasts and payout ratios and, (iii) analysts' dividend forecasts. While each of these methods offers valuable insights into future dividend payments, the first two approaches are contingent on assumptions that may not hold true in most situations. Moreover, analysts tend to generate less informative and more biased forecasts for firms with complex information structures.

To overcome these limitations, this study adopts an approach by integrating the information extracted from these three measures, along with other relevant data not encompassed within these measures, into a single forecast generation process. We achieve this by employing a non-parametric tree-based machine learning approach to forecast dividends. We demonstrate that our machine learning algorithm produces more accurate forecasts and outperforms the alternative methods commonly

used in the literature. Furthermore, we highlight that machine learning-generated dividend forecasts remain unaffected by firm complexity, while analysts produce less accurate dividend forecasts for firms with complex structures. As a practical application, we use our machine learning-generated dividend forecasts as input in the Dividend Discount Model to compute a forward-looking expected return, often referred to as the implied cost of capital. Our new expected return proxy contains more predictive power about future realized returns compared to traditional methods. This superiority holds true both in-sample and out-of-sample.

Our results have academic and practical implications. For practical applications, our findings suggest that the economic advantages of using machine learning are more pronounced in complex information environments. Given the ongoing exponential growth in big data,[16] this result implies that investors should increasingly rely on machine learning for more accurate forecasts. On the academic front, our study suggests that expected return measures derived from machine learning forecasts can encompass richer information about future realized returns, further emphasizing the importance of exploring machine learning applications to better understand the dynamics of the risk-return relationship in the asset pricing literature.

[16] McKinsey Global Institute: Big data: The next frontier for innovation, competition, and productivity.

References

Allen, F., Bernardo, A. E., and Welch, I. (2000). A theory of dividends based on tax clienteles. *The journal of finance*, 55(6):2499–2536.

Allen, F. and Michaely, R. (2003). Payout policy. *Handbook of the Economics of Finance*, 1:337–429.

Barinov, A., Park, S. S., and Yıldızhan, Ç. (2022). Firm complexity and post-earnings announcement drift. *Review of Accounting Studies*, pages 1–53.

Bartov, E., Givoly, D., and Hayn, C. (2002). The rewards to meeting or beating earnings expectations. *Journal of accounting and economics*, 33(2):173–204.

Benamar, H., Foucault, T., and Vega, C. (2021). Demand for information, uncertainty, and the response of us treasury securities to news. *The Review of Financial Studies*, 34(7):3403–3455.

Bilinski, P. and Bradshaw, M. T. (2015). Analyst Dividend Forecasts and Their Usefulness to Investors: International Evidence. *SSRN Electronic Journal*.

Bilinski, P. and Bradshaw, M. T. (2022). Analyst dividend forecasts and their usefulness to investors. *The Accounting Review*, 97(4):75–104.

Brav, A., Graham, J. R., Harvey, C. R., and Michaely, R. (2005). Payout policy in the 21st century$. *Journal of Financial Economics*, page 45.

Brown, L. D., Call, A. C., Clement, M. B., and Sharp, N. Y. (2015). Inside the "black box" of sell-side financial analysts. *Journal of Accounting Research*, 53(1):1–47.

Campbell, J. Y. and Shiller, R. J. (1988). The dividend-price ratio and expectations of future dividends and discount factors. *The Review of Financial Studies*, 1(3):195–228.

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

Chalfin, A., Danieli, O., Hillis, A., Jelveh, Z., Luca, M., Ludwig, J., and Mullainathan, S. (2016). Productivity and selection of human capital with machine learning. *American Economic Review*, 106(5):124–127.

Chen, Q. and Jiang, W. (2006). Analysts' weighting of private and public information. *The Review of financial studies*, 19(1):319–355.

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

De La O, R. and Myers, S. (2021). Subjective Cash Flow and Discount Rate Expectations. *The Journal of Finance*, 76(3):1339–1387.

de Silva, T. and Thesmar, D. (2021). Noise in expectations: Evidence from analyst forecasts. Working Paper 28963, National Bureau of Economic Research.

Dessaint, O., Foucault, T., and Frésard, L. (2023). Does alternative data improve financial forecasting? the horizon effect. *Journal of Finance*, Forthcoming.

Dugast, J. and Foucault, T. (2018). Data abundance and asset price informativeness. *Journal of Financial economics*, 130(2):367–391.

Dugast, J. and Foucault, T. (2023). Equilibrium data mining and data abundance. *Journal of Finance*, Forthcoming.

Easton, P. D. and Monahan, S. J. (2005). An evaluation of accounting-based measures of expected returns. *The Accounting Review*, 80(2):501–538.

Ely, K. M. and Mande, V. (1996). The interdependent use of earnings and dividends in financial analysts' earnings forecasts. *Contemporary Accounting Research*, 13(2):435–456.

Fama, E. F. and French, K. R. (1988). Dividend yields and expected stock returns. *Journal of financial economics*, 22(1):3–25.

Fama, E. F. and MacBeth, J. D. (1973). Risk, return, and equilibrium: Empirical tests. *Journal of political economy*, 81(3):607–636.

Farre-Mensa, J., Michaely, R., and Schmalz, M. (2014). Payout policy. *Annu. Rev. Financ. Econ.*, 6(1):75–134.

Floyd, E., Li, N., and Skinner, D. J. (2015). Payout policy through the financial crisis: The growth of repurchases and the resilience of dividends. *Journal of Financial Economics*, 118(2):299–316.

Francis, J. and Philbrick, D. (1993). Analysts' decisions as products of a multi-task environment. *Journal of Accounting Research*, 31(2):216–230.

Gebhardt, W. R., Lee, C. M., and Swaminathan, B. (2001). Toward an implied cost of capital. *Journal of accounting research*, 39(1):135–176.

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies*, 33(5):2223–2273.

Guay, W., Kothari, S., and Shu, S. (2011). Properties of implied cost of capital using analysts' forecasts. *Australian Journal of Management*, 36(2):125–149.

Hilary, G. and Hsu, C. (2013). Analyst forecast consistency. *the Journal of Finance*, 68(1):271–297.

Hoitash, R. and Hoitash, U. (2018). Measuring accounting reporting complexity with xbrl. *The Accounting Review*, 93(1):259–287.

Hoitash, R., Hoitash, U., and Yezegel, A. (2021). Can sell-side analysts' experience, expertise and qualifications help mitigate the adverse effects of accounting reporting complexity? *Review of Quantitative Finance and Accounting*, 57:859–897.

Hommel, N., Landier, A., and Thesmar, D. (2023). Corporate valuation: An empirical comparison of discounting methods. Working Paper 30898, National Bureau of Economic Research.

Hou, K., van Dijk, M. A., and Zhang, Y. (2012). The implied cost of capital: A new approach. *Journal of Accounting and Economics*, 53(3):504–526.

Huang, D. and Kilic, M. (2019). Gold, platinum, and expected stock returns. *Journal of Financial Economics*, 132(3):50–75.

James, G., Witten, D., Hastie, T., Tibshirani, R., et al. (2013). *An introduction to statistical learning*, volume 112. Springer.

Kane, A., Lee, Y. K., and Marcus, A. (1984). Earnings and dividend announcements: is there a corroboration effect? *The Journal of Finance*, 39(4):1091–1099.

Kelly, B. T., Malamud, S., and Zhou, K. (2022). The virtue of complexity in return prediction. Working Paper 30217, National Bureau of Economic Research.

Lacerda, F. and Santa-Clara, P. (2010). Forecasting dividend growth to better predict returns. *Manuscript Universidade Nova de Lisboa*.

Leary, M. T. and Michaely, R. (2011). Determinants of Dividend Smoothing: Empirical Evidence. *Review of Financial Studies*, 24(10):3197–3249.

Lee, C., Ng, D., and Swaminathan, B. (2009). Testing international asset pricing models using implied costs of capital. *The Journal of Financial and Quantitative Analysis*, 44(2):307–335.

Lee, C., So, E., and Wang, C. (2010). Evaluating implied cost of capital estimates. *SSRN eLibrary*, 6:51.

Lee, C. M. C., So, E. C., and Wang, C. C. Y. (2020). Evaluating Firm-Level Expected-Return Proxies: Implications for Estimating Treatment Effects. *The Review of Financial Studies*, 34(4):1907–1951.

Lehavy, R., Li, F., and Merkley, K. (2011). The effect of annual report readability on analyst following and the properties of their earnings forecasts. *The Accounting Review*, 86(3):1087–1115.

Li, Y., Ng, D. T., and Swaminathan, B. (2012). Predicting time-varying value premium using the implied cost of capital: Implications for countercyclical risk, mispricing and style investing. *Available at SSRN*, 2082108.

Li, Y., Ng, D. T., and Swaminathan, B. (2013). Predicting market returns using aggregate implied cost of capital. *Journal of Financial Economics*, 110(2):419–436.

Lintner, J. (1956). Distribution of incomes of corporations among dividends, retained earnings, and taxes. *The American economic review*, 46(2):97–113.

Loughran, T. and McDonald, B. (2020). Measuring firm complexity. *Journal of Financial and Quantitative Analysis*, pages 1–55.

Michaely, R. and Moin, A. (2022). Disappearing and reappearing dividends. *Journal of Financial Economics*, 143(1):207–226.

Miller, M. H. and Rock, K. (1985). Dividend policy under asymmetric information. *The Journal of Finance*, 40(4):1031–1051.

Newey, W. K. and West, K. D. (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica: Journal of the Econometric Society*, pages 703–708.

Pástor, L., Sinha, M., and Swaminathan, B. (2008). Estimating the intertemporal risk–return tradeoff using the implied cost of capital. *The Journal of Finance*, 63(6):2859–2897.

So, E. C. (2013). A new approach to predicting analyst forecast errors: Do investors overweight analyst forecasts? *Journal of Financial Economics*, 108(3):615–640.

van Binsbergen, J. H., Han, X., and Lopez-Lira, A. (2022). Man versus Machine Learning: The Term Structure of Earnings Expectations and Conditional Biases. *The Review of Financial Studies*, 36(6):2361–2396.

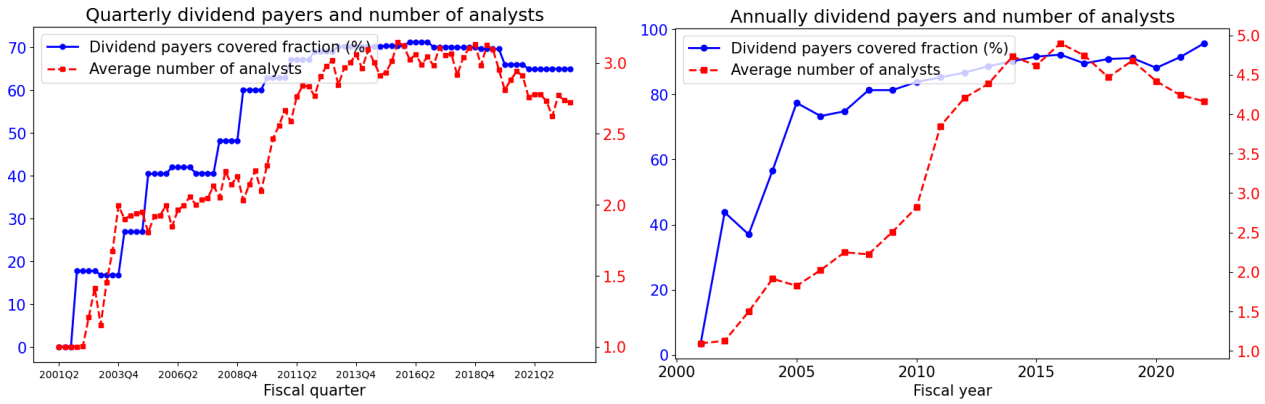
Welch, I. and Goyal, A. (2008). A Comprehensive Look at The Empirical Performance of Equity Premium Prediction. *Review of Financial Studies*, 21(4):1455–1508.

Yagan, D. (2015). Capital Tax Reform and the Real Economy: The Effects of the 2003 Dividend Tax Cut. *American Economic Review*, 105(12):3531–3563.

Yin, L. and Nie, J. (2021). Adjusted dividend-price ratios and stock return predictability: Evidence from china. *International Review of Financial Analysis*, 73:101618.

Zwiebel, J. (1996). Dynamic capital structure under managerial entrenchment. *The American Economic Review*, 86(5):1197–1215.

Figure 1: Analyst forecast of DPS coverage

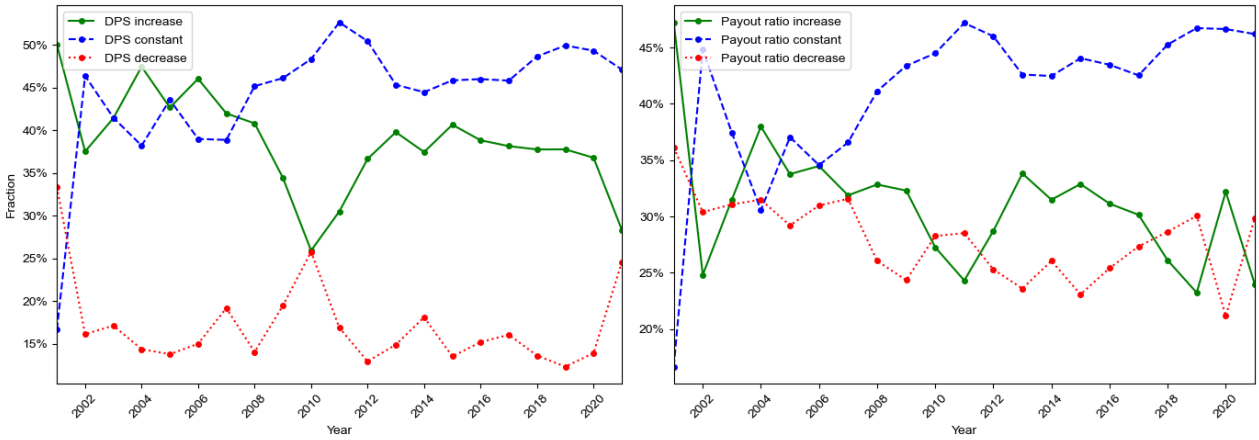


(a) analyst forecast quarterly coverage

(b) analyst forecast annually coverage

Note: This figure displays the proportion of quarterly (a) and annually (b) dividend payers covered by analyst forecasts for the US companies from 2001 to 2022. The blue line with circular markers represents the dividend payers covered fraction, while the red line with square markers represents the average number of analysts report for each firm-level forecast. The data of actual and analyst forecast of fiscal annual and fiscal quarter dividend is from I/B/E/S.

Figure 2: Non-sticky dividend payout

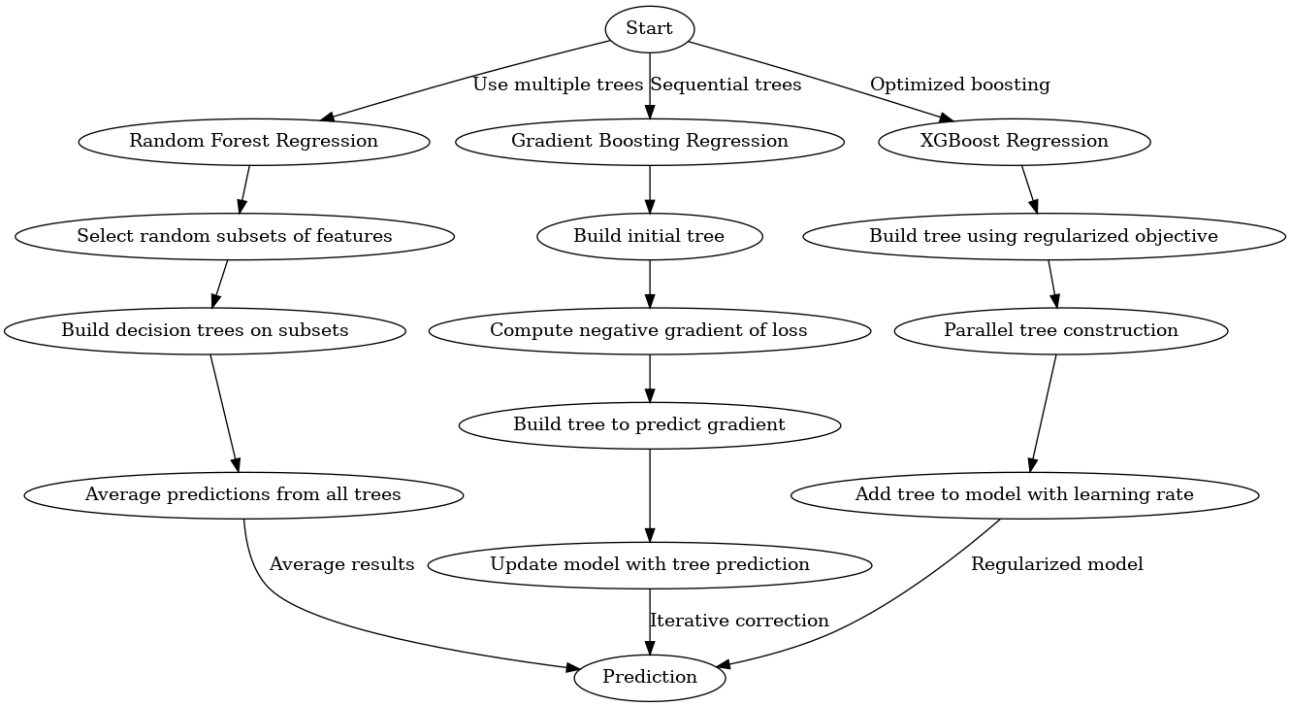


(a) Fractions of dividend payout behaviors

(b) Fractions of dividend payout ratio behaviors

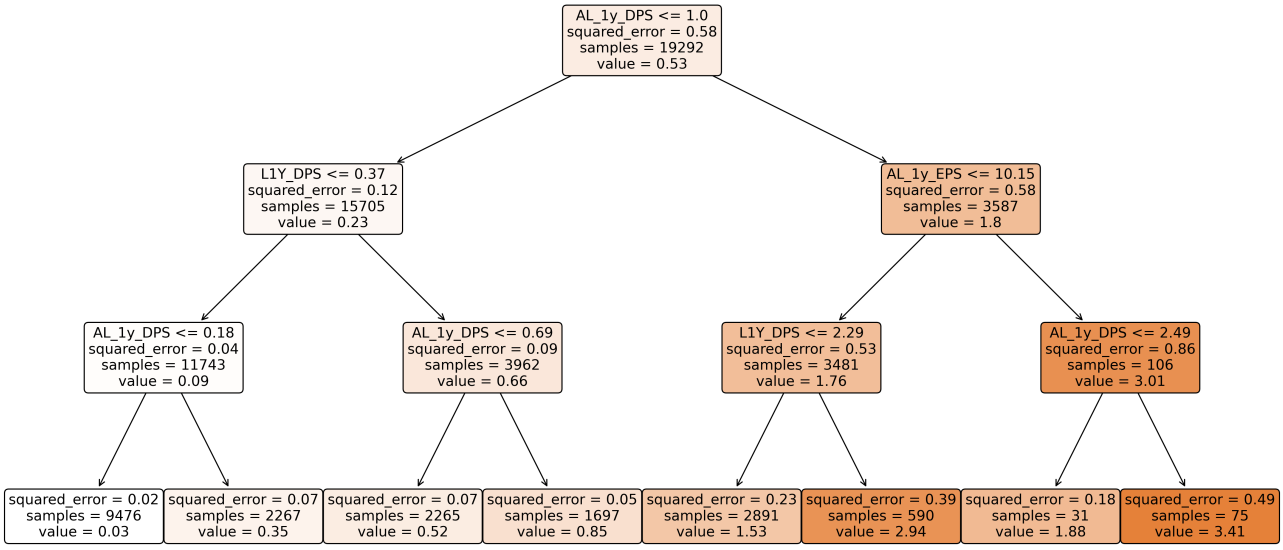
Note: This figure shows the proportion of dividend paying firms that increase, maintain, and decrease their dividend payout ratio [Panel (a)], and dividend payout ratio [Panel (b)], compared to the previous fiscal year. The payout ratio is calculated by the actual fiscal annual dividend per share (DPS) divided by earnings per share (EPS). The actual fiscal annual data is obtained from I/B/E/S.

Figure 3: Flowchart of three tree-based regression models



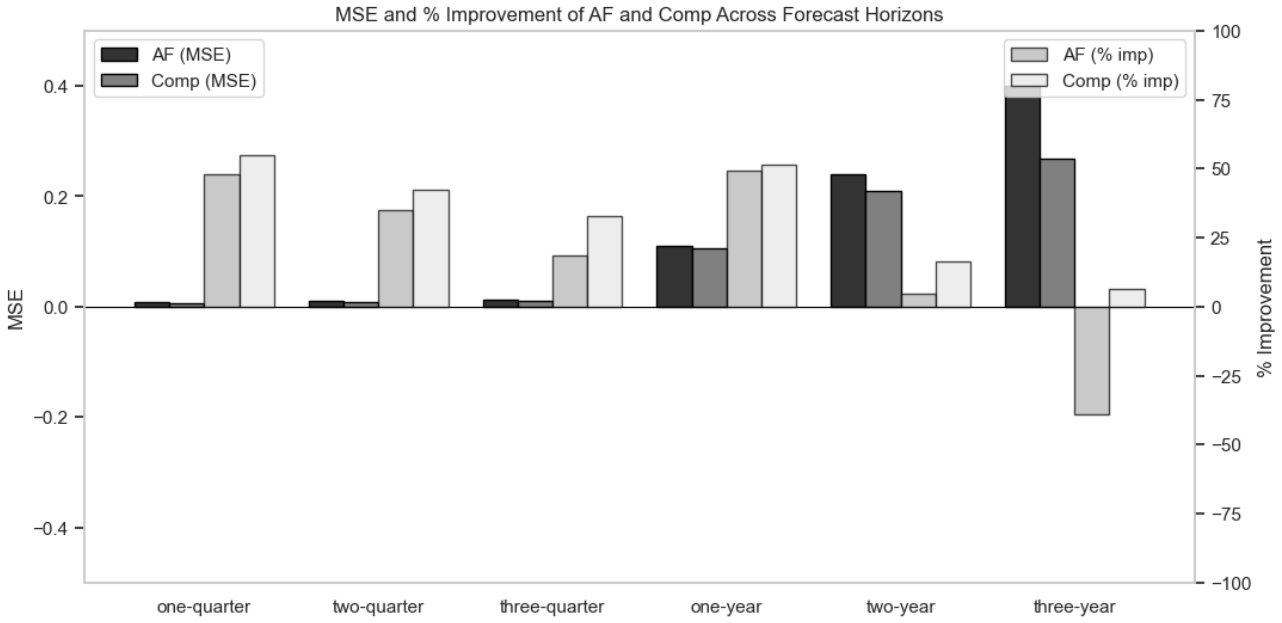
Note: This flowchart illustrates the methodologies employed by three advanced regression algorithms: Random Forest (RF) Regression, Gradient Boosting (GB) Regression, and Extreme Gradient Boosting (XGB) Regression. It provides a visual representation of the key procedural steps involved in the model-building process, leading to the final prediction output.

Figure 4: Example of single regression decision tree



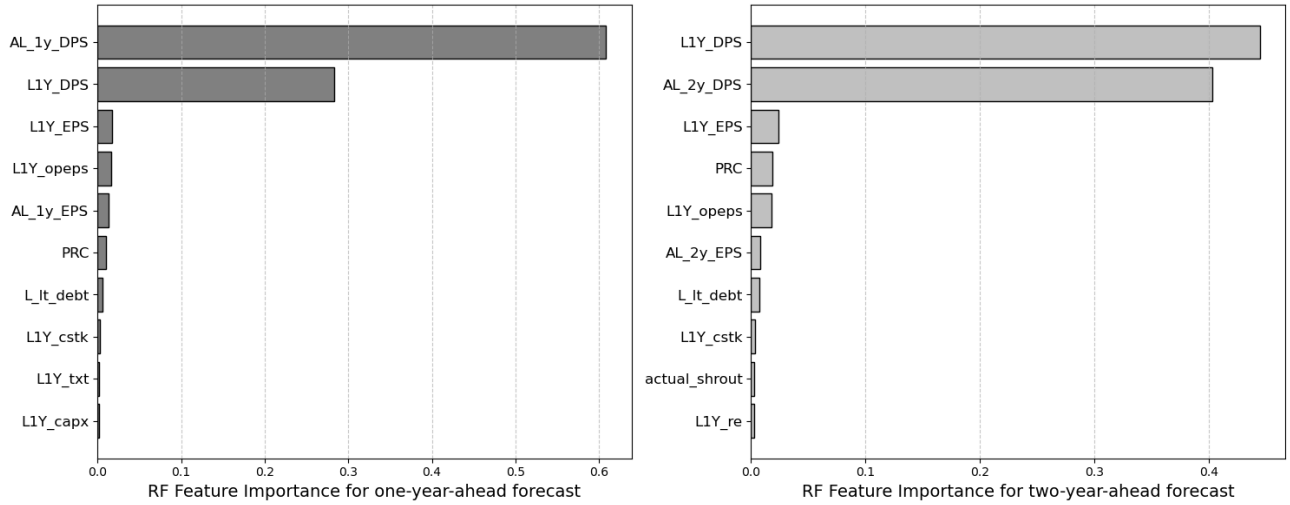
Note: The figure illustrates an example of a regression tree extracted from the Random Forest Regression model for one-year-ahead DPS forecasts. Each node in the tree contains the splitting condition, splitting criteria measured by Mean Squared Error (MSE), the number of observations, and the final output forecast value. The darker-colored nodes indicate areas of the tree with low impurity, signifying that the samples within these nodes have similar target values, resulting in more homogeneous predictions for these samples. AL_1y_DPS and AL_1y_EPS donate the analyst forecast of DPS and EPS, and L1Y_DPS represents the lagged actual DPS payout.

Figure 5: Forecast error of composite machine learning and analyst forecasts



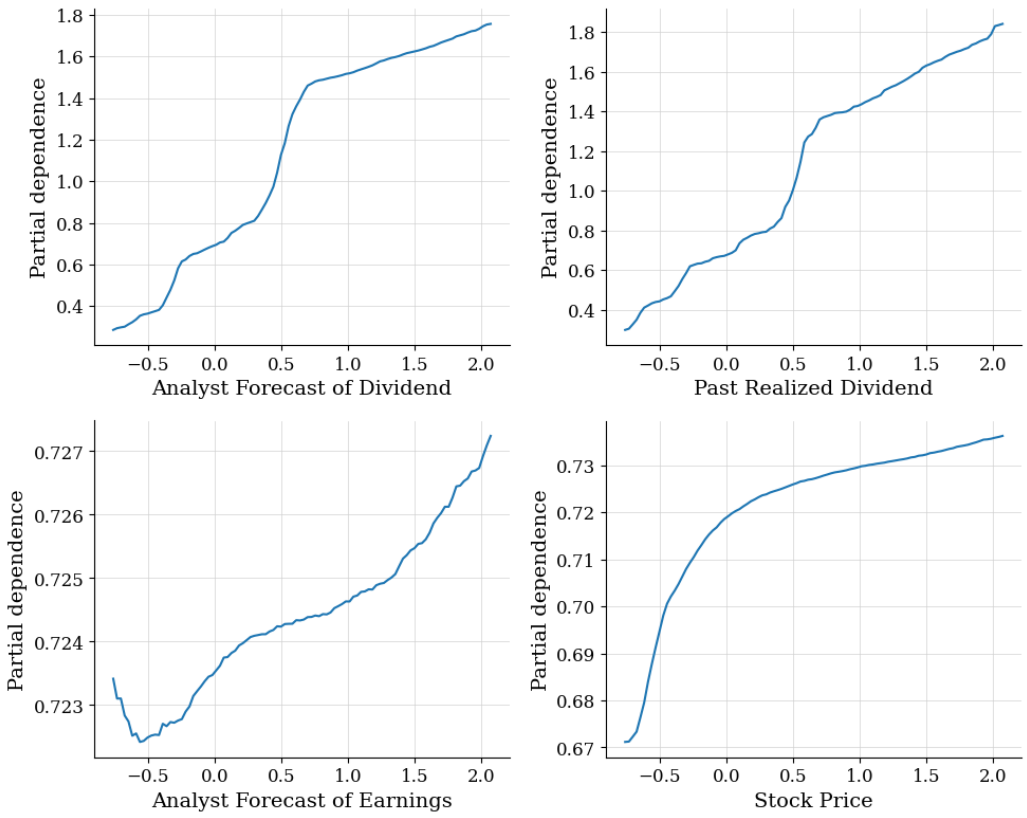
Note: This figure presents a comparison of forecast accuracy between the composite machine learning and analyst forecasts for DPS across various forecasting horizons. Accuracy is assessed using both Mean Squared Error (MSE) and the improvement in MSE relative to the naive forecast, which relies on lagged actual DPS values.

Figure 6: Feature Importance from Random Forest Forecast



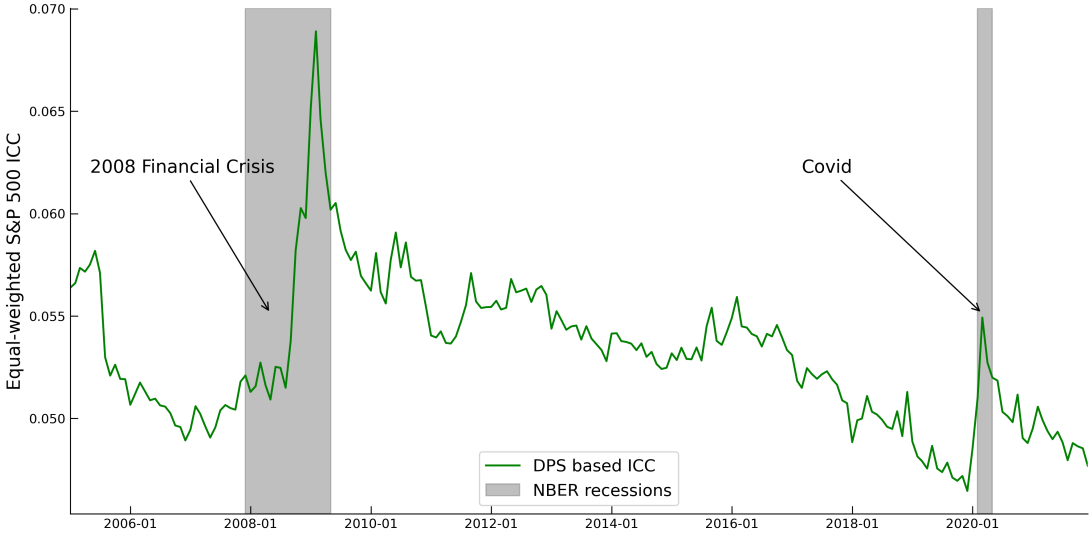
Note: This figure showcases the top 10 most influential features in one and two-year ahead forecasts, as determined by the RF method. The significance of each feature is measured by the reduction in impurity as the feature is incorporated into the forecasting model. All importance scores have been normalized to ensure a cumulative sum of one. The prefix 'AL' implies analyst forecasts, 'L1Y' denotes variables realized during the last fiscal year, and 'L' represents variables realized during the last fiscal quarter.

Figure 7: Nonlinearity between future actual DPS and key features.



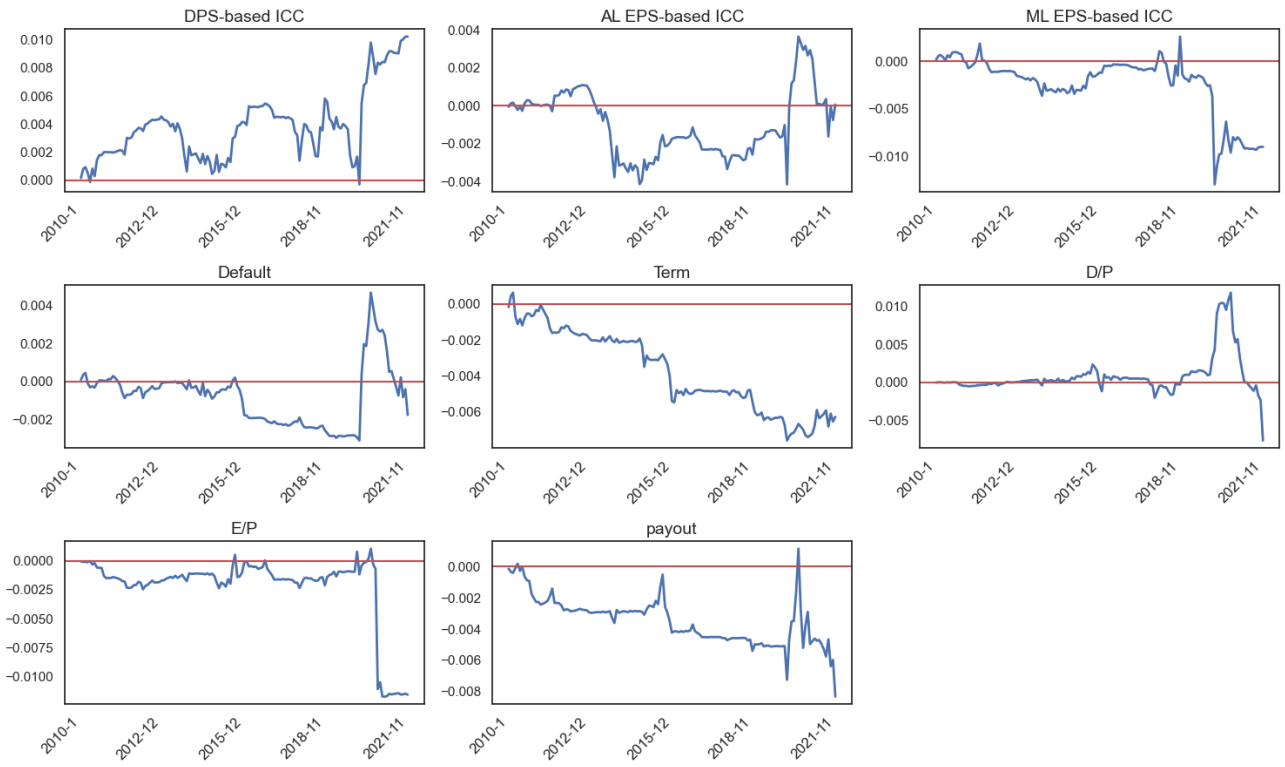
Note: This figure presents Partial Dependence Plots (PDPs) derived from a two-year-ahead DPS forecast using RF regression. The forecasting model utilizes the standardized predictors described earlier. The time period covered in the analysis spans from 2003 to 2022.

Figure 8: Time-series of the S&P 500 aggregate ICC



Note: The figure plots the aggregate ICC computed as the value-weighted average of ICC values for all included S&P 500 constituents. The data covers the period from January 2005 to December 2021, with ICC expressed in an annualized format. These ICC values are derived from our machine learning forecasts of DPS, encompassing predictions up to three years ahead. The shaded regions on the graph correspond to NBER recession periods, which include the 2008 financial crisis and the 2020 COVID-19 outbreak.

Figure 9: Difference of cumulative squared error between historical average and ERP.



Note: The figure displays the difference of cumulative forecast squared error of value-weighted S&P 500 return between the benchmark forecast using the historical average return and forecasts using the tested ERP, including: EPS and DPS-based ICC, default spread ($Default$), term spread ($Term$), dividend-price ratio (D/P), earnings-price ratio (E/P), and log payout ratio (py). The compare time period spans from January 2010 to December 2021. The out-of-sample forecasts are obtained with a 60-month rolling window method.

Table 1: Variables Utilized in Constructing Machine Learning Forecasts for DPS.
The table presents an overview of the three information sets utilized in constructing the machine learning forecast. Analyst forecasts, as well as realized DPS and EPS data, are sourced from I/B/E/S. Quarterly and annual firm fundamentals data are extracted from the WRDS CRSP/Compustat Merged database. Monthly stock price, return, and trading volume data are obtained from CRSP, while macroeconomic indicators are sourced from FRED.

Predictors	Category	Description
$X_{i,t}^{AL}$	Analyst forecasts	1. Analyst forecast of DPS and EPS 2. Analyst forecast of long-run EPS growth rate averaged by industries.
$X_{i,t}^{Fin}$	Firm fundamentals	1. Realized last annually and quarterly DPS and EPS. 2. 58 quarterly financial fundamentals and calculated financial ratios. 3. Monthly stock price, return, and number of shares.
X_t^{Macro}	Macroeconomic indicators	GDP Growth, industrial production index (IPT) growth, consumption expenditure growth, and unemployment rate.

Table 2: Summary Statistics and Accuracy Evaluation of DPS Forecasts.

This table presents summary statistics for actual DPS values and various DPS forecast results in the first panel, followed by accuracy evaluation in the second panel. The forecast period spans from January 2004 to December 2022. *Actual* represents the actual DPS values, while *AF* and $EPS * payout$ denote the analyst's DPS forecast and the analyst's forecast of EPS multiplied by the most recent year's payout ratio, respectively. *RF*, *GB*, and *XGB* correspond to forecasts generated using three tree-based machine learning regression methods: Random Forest, Gradient Boost Trees, and XGBoost Trees. *Comp* indicates the composite machine learning forecast, computed as the average of *RF* and *GB*. In the second panel, MSE represents the Mean Squared Error, calculated as the mean squared difference between actual and forecasted DPS values. % imp signifies the percentage reduction in MSE achieved by the forecasting methods compared to the naive forecast using the lagged one-period actual DPS. A negative sign ('-') indicates a reduction exceeding 100%.

Summary statistics	one-quarter		two-quarter		three-quarter		one-year		two-year		three-year	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
Actual	0.204	0.248	0.210	0.254	0.216	0.260	0.678	0.945	0.730	1.014	0.815	1.111
AF	0.204	0.244	0.208	0.245	0.214	0.251	0.670	0.926	0.721	0.974	0.823	1.089
$EPS * payout$	0.175	0.248	0.187	0.263	0.197	0.279	0.613	0.955	0.693	1.024	0.839	1.249
RF	0.203	0.228	0.209	0.228	0.215	0.232	0.672	0.863	0.724	0.877	0.808	0.939
GB	0.203	0.226	0.209	0.230	0.215	0.234	0.674	0.858	0.725	0.883	0.810	0.946
XGB	0.202	0.230	0.209	0.232	0.215	0.236	0.675	0.879	0.726	0.897	0.814	0.968
Comp	0.203	0.227	0.209	0.228	0.215	0.233	0.673	0.859	0.725	0.878	0.809	0.939
Obs.	385847		372989		352092		534255		475311		316083	
N. of firms	5173		5095		4994		7213		6520		5047	
Accuracy evaluation	one-quarter		two-quarter		three-quarter		one-year		two-year		three-year	
	MSE	% imp	MSE	% imp	MSE	% imp	MSE	% imp	MSE	% imp	MSE	% imp
DPS_{t-1}	0.015		0.016		0.016		0.217		0.251		0.287	
AF	0.008	47.92	0.010	34.81	0.013	18.39	0.110	49.18	0.239	4.50	0.400	-39.20
$EPS * payout$	0.029	-89.53	0.038	-	0.044	-	0.387	-78.11	0.445	-77.65	0.769	-
RF	0.007	55.01	0.009	42.55	0.011	33.04	0.105	53.68	0.211	15.64	0.271	5.83
GB	0.007	52.53	0.009	40.42	0.011	31.04	0.107	50.80	0.214	14.64	0.277	3.44
XGB	0.007	52.06	0.010	40.24	0.011	30.26	0.113	48.03	0.225	10.33	0.287	0.05
Comp	0.007	54.81	0.009	42.45	0.011	33.01	0.106	51.38	0.210	16.30	0.268	6.58

Table 3: Firm Complexity and Forecast Errors.

Panel A of this table presents the time-series averages of squared errors for analyst forecasts and machine learning forecasts within five portfolios sorted by firm complexity. Firm complexity is quantified by ARC/100, representing one hundred units of monetary items disclosed. The Mean Squared Error (MSE) for each group is calculated as the value-weighted average of all individuals' three-year-ahead DPS forecast errors within the respective year. Panel B provides time-series average results for annual cross-sectional regression tests that examine the relationship between forecast errors and complexity. In Column (A), the regression includes ARC as the sole explanatory variable, while Column (B) introduces additional controls, encompassing firm fundamentals such as the logarithms of size and book-to-market ratio, return on assets (ROA), debt-equity ratio (d/e), dividend payout ratio (payout), number of employees (emp), and return volatility (ret_vol). Furthermore, two other firm complexity measures, *Conglo* and *GeoMulti*, are included. *Conglo* is a dummy variable that takes a value of one if a firm has multiple business segments in the same year, while *GeoMulti* is one if a firm generates sales from multiple geographic segments in the same year. Ind_FE indicates the including of industry fixed effects. The *t*-statistics are computed using the Fama-Macbeth method for the average slopes. The data spans from 2009 to 2021. *, **, and *** indicate significance at 10%, 5%, and 1% levels, respectively.

Panel A: portfolio sort	1	2	3	4	5	C-S
AL_MSE	0.202	0.219	0.239	0.246	0.290	0.088*
t-stat						1.920
ML_MSE	0.143	0.133	0.153	0.171	0.175	0.032
t-stat						1.204

Panel B: Regression test	Analyst forecast		Machine learning forecast	
	(A)	(B)	(A)	(B)
ARC	0.039***	0.035***	0.004	-0.003
t-stat	4.865	3.772	0.754	-0.507
Conglo		-0.028		-0.024
t-stat		-1.140		-1.323
GeoMulti		-0.057**		-0.022
t-stat		-2.095		-1.162
Size		0.039***		0.033***
t-stat		4.140		4.769
b/m		-0.004		0.000
t-stat		-0.293		-0.030
ROA		0.324***		0.206***
t-stat		10.29		7.921
d/e		0.001***		0.000**
t-stat		3.665		1.718
payout		0.004***		0.002
t-stat		11.28		6.109
emp		0.000		0.000
t-stat		-0.594		-0.166
ret_vol		0.364***		0.121
t-stat		2.434		1.218
Intercept	0.311***	-0.243*	0.280***	-0.177*
t-stat	9.475	-1.899	11.005	-1.931
Ind_FE	Y	Y	Y	Y
$R\hat{2}$	3.618%	5.217%	2.798%	4.342%

Table 4: ICC calculation approaches.
The table presents a comparison between the ICC imputed from the conventional EPS-based method (denoted as r_e) and the DPS-based ICC obtained from a simplified Gordon growth model (denoted as r_d).

ICC	Assumption	Computation model
r_e	1. The growth of earnings after year 2 mean-reverts exponentially to GDP growth rate g. 2. The plowback rate follows the iterated process of $b_{t+k} = b_{t+k-1} - (b_1 - b)/T$. 3. In steady-state plowback rate b times the return on new investment equals the steady-state earnings growth rate.	$P_t = \sum_{k=1}^{T=15} \frac{E_t[EPS_{t+k}] \times (1 - b_{t+k})}{(1 + r_{e,t})^k} + \frac{E_t[EPS_{t+16}]}{r_{e,t}(1 + r_{e,t})^T}$
r_d	Dividend after year 2 will grow perpetually at the average GDP growth rate.	$P_t = \sum_{k=1}^{T=2} \frac{E_t[DPS_{t+k}]}{(1 + r_{d,t})^k} + \frac{E_t[DPS_{t+3}]}{(r_{d,t} - g)(1 + r_{d,t})^T}$

Table 5: Summary statistics of S&P 500 return and predictors.

This table presents summary statistics for various variables used in our in-sample prediction tests covering the period from January 2005 to December 2021, including S&P 500 returns. The predictors encompass EPS (both $r_{e,AL}$ and $r_{e,ML}$), DPS-based ICC (r_d), calculated using both equal and value-weighted methods, default spread ($Default$), term spread ($Term$), dividend-price ratio (D/P), earnings-price ratio (E/P), and the natural logarithm of the payout ratio (py). All variables, except for payout, are expressed as annualized percentages. The table also includes columns for AR(1) and AR(12), indicating the autocorrelation coefficients at one and twelve lags, respectively. Additionally, ADF represents the augmented Dickey-Fuller test statistic, and the accompanying p-value is provided.

Variable	mean	std	Min.	Max.	AR(1)	AR(12)	ADF	p-value
$ewret$	0.121	0.595	−2.491	2.218	0.076	0.029	−13.115	0.000
ew_r_d	0.052	0.003	0.046	0.071	0.922	0.179	−3.542	0.007
$ew\_r_{e,AL}$	0.088	0.011	0.072	0.128	0.957	0.548	−2.155	0.223
$ew\_r_{e,ML}$	0.088	0.014	0.068	0.136	0.964	0.361	−2.565	0.100
$vwret$	0.901	0.509	−2.199	1.453	0.078	0.036	−13.111	0.000
vw_r_d	0.053	0.004	0.046	0.069	0.938	0.423	−2.769	0.063
$vw\_r_{e,AL}$	0.088	0.011	0.071	0.126	0.960	0.630	−2.049	0.265
$vw\_r_{e,ML}$	0.086	0.014	0.065	0.137	0.961	0.383	−2.430	0.133
$Default$	0.126	0.055	0.066	0.406	0.953	0.081	−3.743	0.004
$Term$	0.012	0.009	−0.001	0.028	0.985	0.682	−1.727	0.417
D/P	0.020	0.003	0.013	0.036	0.950	0.124	−3.305	0.015
E/P	0.048	0.013	0.008	0.074	0.982	0.315	−2.462	0.125
py	−0.362	0.210	−0.540	0.599	0.980	0.083	−4.113	0.001

Table 6: Predicting future realized returns by using expected returns: univariate anaysis.
This table presents the results of predictive regression analysis for equal (left panel) and value-weighted (right panel) S&P 500 returns, covering the period from January 2005 to December 2021. The analysis is conducted at forecast horizons of 1, 6, and 12 months.
The regression is: $\sum_{h=1}^{H}(r_{t+h})/H = \beta_0 + \beta_1 \times M_t + \epsilon_{t+h}$.
where r_{t+h} represents the continuously compounded monthly returns, either equal-weighted or value-weighted, in excess of the compounded three-month T-bill rate. M_t donates the vector of predictors, including EPS ($r_{e,AL}, r_{e,ML}$) and DPS-based ICC (r_d), computed using equal or value-weighted methods, default spread ($Default$), term spread ($Term$), dividend-price ratio (D/P), earnings-price ratio (E/P), and log payout ratio ($payout$). The reported values under "t-stat" represent the Newey and West (1987) t-statistic, calculated with the number of lags equal to the forecast horizon length. *, **, and *** indicate significance at 10%, 5%, and 1% levels, respectively.

		Equal-weighted			Value-weighted		
variable	Horizon	β_1	t-stat	R-sq (%)	β_1	t-stat	R-sq (%)
r_d	1	3.145*	1.822	4.34	2.191*	1.654	2.66
	6	3.328***	7.479	23.76	2.389***	5.373	16.01
	12	2.822***	6.748	36.02	2.018***	3.801	22.51
$r_{e,AL}$	1	0.270	0.490	0.35	0.074	0.175	0.04
	6	0.401	1.129	3.80	0.106	0.340	0.37
	12	0.503***	3.540	12.28	0.204	1.569	2.65
$r_{e,ML}$	1	0.293	0.731	0.66	0.159	0.521	0.26
	6	0.445*	1.888	7.46	0.206	1.004	2.26
	12	0.422***	3.536	13.86	0.159	1.529	2.63
$Default$	1	0.029	0.018	0.00	-0.505	-0.417	0.29
	6	1.021	1.359	4.11	0.331	0.490	0.63
	12	1.162***	4.030	11.01	0.549**	2.054	3.38
$Term$	1	0.283	0.706	0.23	0.090	0.267	0.03
	6	0.273	0.987	1.06	0.088	0.414	0.16
	12	0.339	1.549	3.44	0.160	0.926	1.05
D/P	1	1.106	0.526	0.53	0.241	0.151	0.04
	6	1.876*	1.860	6.72	0.907	1.030	2.30
	12	2.102**	5.419	16.12	1.228***	3.578	7.57
E/P	1	-0.101	-0.250	0.07	0.023	0.073	0.00
	6	-0.281	-0.999	2.54	-0.140	-0.641	0.93
	12	-0.258	-1.379	4.32	-0.121	-0.875	1.32
py	1	0.022	0.776	0.84	0.009	0.393	0.18
	6	0.031***	2.713	7.90	0.017*	1.821	3.74
	12	0.025***	3.983	11.05	0.014***	2.843	4.58

Table 7: Predicting future realized returns by using expected returns: bivariate anaysis.
This table presents bivariate predictive regression results for one-month-ahead S&P 500 returns, covering the period from January 2005 to December 2021. The top and bottom panels display results for equal-weighted and value-weighted returns, respectively. The predictors considered include $r_{e,AL}$, $r_{e,ML}$, and r_d. We also include two additional predictors: D/P and *Term*. The first and second column reports the estimated coefficient and corresponding Newey and West (1987) t-statistic with one-lag for the fixed explanatory variable r_d in different regressions.
The regression is: $r_{t+1} = \beta_0 + \beta_1 \times r_{d,t} + \beta_2 \times N_t + \epsilon_{t+h}$, where $N_t \in \{r_{e,AL}, r_{e,ML}, D/P, E/P, payout\}$. *, **, and *** indicate significance at 10%, 5%, and 1% levels, respectively.

Equal-Weighted return prediction										
r_d		$r_{e,AL}$		$r_{e,ML}$		D/P		*Term*		
β_1	t-stat	β_2	t-stat	β_2	t-stat	β_2	t-stat	β_2	t-stat	R-sq
7.326***	2.653	-1.536*	-1.853							7.18
5.514**	2.216			-0.729	-1.358					5.01
5.404***	3.310					-3.006	-1.380			5.08
4.002*	1.805							-0.591	-1.043	4.12
Value-Weighted return prediction										
6.047***	2.641	-1.373**	-2.045							5.76
3.593**	2.030			-0.442	-1.142					2.67
4.573***	3.057					3.033	-1.604			4.20
3.455*	1.778							-0.726	-1.371	2.98

Table 8: Out-of-sample predictability test.

This table presents out-of-sample predictive test results for one-month-ahead S&P 500 returns, covering the period from January 2005 to December 2021. The forecast period spans from January 2010 (2015) for a 60 (120) months estimation window and ends at December 2021. The top and bottom panels display results for equal-weighted and value-weighted returns, respectively. The left and right panels indicate an estimation strategy with a rolling window and expanding window, respectively. Definitions of the tested ERP are the same as in the above tables. The first column reports out-of-sample R square R^2_{OS} calculated by: $R^2_{OS} = 1 - \frac{\sum_{k=1}^{T-m}(r_{m+k} - \hat{r}_{m+k})^2}{\sum_{k=1}^{T-m}(r_{m+k} - \bar{r}_{m+k})^2}$. The second column p-val stands for the p-value from the one-side test results using the adjusted-MSPE statistic regress on a constant, calculated as: $f_{t+1} = (r_{t+1} - \bar{r}_{t+1})^2 - [(r_{t+1} - \hat{r}_{t+1})^2 - (\bar{r}_{t+1} - \hat{r}_{t+1})^2]$. *, **, and *** indicate significance at 10%, 5%, and 1% levels, respectively.

Equal-weighted	Rolling window				Expanding window			
	60 months		120 months		60 months		120 months	
	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val
r_d	5.54**	0.013	6.41**	0.020	4.73***	0.008	5.41**	0.024
$r_{e,AL}$	−0.08		1.21	0.159	0.36	0.248	0.28	0.314
$r_{e,ML}$	−6.92		−1.95		0.39	0.261	0.16	0.360
$Default$	−0.88		1.52*	0.091	0.04	0.410	−0.13	
$Term$	−2.86		−2.07		−0.34		−0.83	
D/P	−3.46		−6.17		0.04	0.340	−0.44	
E/P	−7.13		−2.33		−1.44		−0.14	
$payout$	−4.34		−0.41		−0.98		0.72	0.163
Value-weighted	Rolling window				Expanding window			
	60 months		120 months		60 months		120 months	
	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val
r_d	4.36***	0.010	1.06	0.106	3.00**	0.024	1.75	0.129
$r_{e,AL}$	0.03	0.141	−0.03	0.356	−0.23		−0.37	
$r_{e,ML}$	−3.84		−1.91		−0.21		−0.35	
$Default$	−3.23		−6.37		0.16	0.289	−0.30	
$Term$	−4.92		−1.87		−0.68		−0.89	
D/P	−3.55		−0.38		−0.01		−0.33	
E/P	−0.75		1.61	0.207	−0.64		−0.86	
py	−2.67		−1.97		−0.58		0.33	0.223

Table 9: Evaluation of expected return proxy by variance of measurement errors.
This table presents the measurement error tests for one, six, and twelve-month-ahead S&P 500 returns, covering the period from January 2005 to December 2021. The left and right panels display results for equal-weighted and value-weighted returns, respectively. Definitions of the tested ERP are the same as in the above tables. The first and second column, $Var(\hat{er}_t)$ and $Cov(r_{t+h}, \hat{er}_t)$,indicates time-series variance and covariance between ERP and realized return, respectively. The third column represents the calculated scaled measurement error variance $SMEV_i$ using: $SMEV_i = var_i(\hat{er}_{i,t}) - 2cov_i(r_{i,t+h}, \hat{er}_{i,t})$. All the values are reported after multiplying 100 as suggested by Lee et al. (2020).

		Equal-weighted			Value-weighted		
ERP	Horizon	$Var(\hat{er}_t)$	$Cov(r_{t+h}, \hat{er}_t)$	$SMEV_i$	$Var(\hat{er}_t)$	$Cov(r_{t+h}, \hat{er}_t)$	$SMEV_i$
r_d	1	0.0011	0.0035	-0.0059	0.0010	0.0022	-0.0034
	6	0.0011	0.0038	-0.0064	0.0010	0.0024	-0.0038
	12	0.0012	0.0033	-0.0054	0.0010	0.0021	-0.0032
$r_{e,AL}$	1	0.0122	0.0033	0.0056	0.0117	0.0009	0.0100
	6	0.0125	0.0050	0.0025	0.0120	0.0013	0.0095
	12	0.0126	0.0063	-0.0001	0.0120	0.0024	0.0071
$r_{e,ML}$	1	0.0195	0.0057	0.0081	0.0189	0.0030	0.0129
	6	0.0199	0.0089	0.0022	0.0194	0.0040	0.0114
	12	0.0201	0.0085	0.0031	0.0195	0.0031	0.0133
$Default$	1	0.0021	0.0001	0.0020	0.0021	-0.0010	0.0042
	6	0.0021	0.0021	-0.0022	0.0021	0.0007	0.0007
	12	0.0021	0.0025	-0.0028	0.0021	0.0012	-0.0002
$Term$	1	0.0074	0.0021	0.0032	0.0074	0.0007	0.0060
	6	0.0075	0.0021	0.0034	0.0075	0.0007	0.0062
	12	0.0078	0.0026	0.0025	0.0078	0.0012	0.0053
D/P	1	0.0011	0.0012	-0.0013	0.0011	0.0003	0.0006
	6	0.0010	0.0019	-0.0028	0.0010	0.0009	-0.0008
	12	0.0009	0.0020	-0.0030	0.0009	0.0012	-0.0014
E/P	1	0.0168	-0.0017	0.0202	0.0168	0.0004	0.0161
	6	0.0171	-0.0048	0.0267	0.0171	-0.0024	0.0219
	12	0.0169	-0.0043	0.0255	0.0169	-0.0020	0.0209
py	1	4.4096	0.0970	4.2157	4.4096	0.0375	4.3346
	6	4.4878	0.1371	4.2136	4.4878	0.0779	4.3320
	12	4.6172	0.1149	4.3874	4.6172	0.0631	4.4911

Table 10: Univariate predictive regression for value and size portfolios

This table reports the univariate predictive regression test results for three value sorted portfolios: low (L), medium (M), and high (H) on the first panel, and three size sorted portfolios: small (S), medium (M), and big (B) on the second panel. The tested period spans from January 2005 to December 2021 and portfolios are formed in June of each year.

The regression is: $\sum_{h=1}^{H}(r_{t+h})/H = \beta_0 + \beta_1 \times r_t + \epsilon_{t+h}$.

We focus on comparing the performance of three ICC $r_{e,AL}$, $r_{e,ML}$, and r_d, computed by value-weighting the individual constituents by capitalization. Columns 't-stat' represents Newey and West (1987) t-statistic with the number of lags equal to the forecast length; 'R-sq(%)' is the prediction R^2 reported in percentage. Rows 'average' stands for the average of estimated beta β_1 and R-sq(%) by different forecast horizons. *, **, and *** indicate significance at 10%, 5%, and 1% levels, respectively.

		r_d			$r_{e,AL}$			$r_{e,ML}$		
	Horizon	β_1	t-stat	R-sq (%)	β_1	t-stat	R-sq (%)	β_1	t-stat	R-sq (%)
Value										
L	1	2.17**	2.09	2.44	0.08	0.18	0.03	0.17	0.56	0.27
	6	2.22***	3.55	12.80	0.07	0.22	0.13	0.18	0.82	1.57
	12	1.95***	2.91	20.32	0.16	0.77	1.33	0.15	1.14	1.97
	average	2.11		11.85	0.10		0.50	0.17		1.27
M	1	2.36	1.45	2.91	0.09	0.20	0.05	0.30	1.01	0.84
	6	3.19***	5.06	24.64	0.28	0.87	2.49	0.51***	3.37	11.93
	12	2.78***	4.27	38.98	0.38***	2.60	9.84	0.44***	4.83	18.53
	average	2.78		22.18	0.25		4.13	0.41		10.43
H	1	1.39	1.22	1.40	0.28	1.09	0.74	0.02	0.07	0.00
	6	2.11***	3.52	15.54	0.46***	2.52	8.76	0.32**	2.16	5.22
	12	1.96***	4.08	28.45	0.54***	4.00	26.18	0.48***	3.84	25.06
	average	1.82		15.13	0.43		11.89	0.27		10.09
Size										
S	1	1.43	1.17	1.26	0.82	1.53	1.42	0.84*	1.90	2.33
	6	1.70*	1.91	8.51	0.92***	2.66	9.14	1.15***	4.52	22.06
	12	1.68***	2.87	16.93	0.94***	3.47	19.81	1.07***	5.88	39.39
	average	1.60		8.90	0.89		10.12	1.02		21.26
M	1	2.36*	1.69	2.42	0.44	0.76	0.54	0.46	1.09	0.99
	6	2.65***	4.61	15.72	0.64*	1.85	6.23	0.79***	3.88	15.73
	12	2.30***	6.16	25.52	0.73***	3.95	17.73	0.70***	5.06	27.35
	average	2.44		14.55	0.61		8.17	0.65		14.69
B	1	2.56*	1.91	3.20	0.06	0.15	0.02	0.21	0.69	0.43
	6	2.99***	5.04	21.13	0.13	0.43	0.56	0.28	1.35	3.60
	12	2.52***	3.92	30.89	0.24	1.79***	3.62	0.23**	2.33	4.97
	average	2.69		18.41	0.15		1.40	0.24		3.00

Internet Appendix for Dividend Forecasts via Machine Learning

Xuesi Wang[1], Leonidas Barbopoulos[1], and Khaladdin Rzayev[1,2]

[1]*Business School, University of Edinburgh, 29 Buccleuch Place, Edinburgh EH8 9JS, UK*

[2]*Koc University, Istanbul, Turkey*

Appendix A. Proofs for Forecast Structure and MSE Decomposition

In this appendix section, we present supplementary proofs for the MSE decomposition results and the proof for Lemma 4. Throughout the proofs, we omit indices that indicate time points and forecast horizons, and we use i to represent the firm i. The proofs for Lemma 1 and Lemma 2 can be found in de Silva and Thesmar (2021).

To evaluate the overall forecast accuracy for all the companies, we calculate the Mean Squared Error (MSE) as the average of squared errors generated by each forecast. The cross-sectional MSE is defined as: $MSE = E[(F_i - y_i)^2]$. Next, we introduce Assumption 1, which serves as the most fundamental assumption.

Assumption 1 *All the conditional forecast component* $[x_i, p_i, \varepsilon_i, \eta_i, b_i]$ *are orthogonal to each other, e.g.,* $E(p_i|x_i) = 0$. *The unpredictable residual* ε_i *has a mean of zero.*

Combining the structures of three forecasts, denoted as F_i^L, F_i^{AL}, and F_i^{Com} as shown in Section 2:

$$F_i^L = x_i + \eta_i,$$

$$F_i^{AL} = x_i + p_i + b_i,$$

$$F_i^{Com} = x_i + \beta(p_i + b_i),$$

and operating under the assumption of orthogonality, we calculate the Mean Squared Error (MSE) for each forecast as follows:

$$\begin{aligned}
MSE^L &= E[(\eta_i - p_i - \varepsilon_i)^2] \\
&= E(\eta_i^2) + E(p_i^2) - 2E[\eta_i]E[p_i] + var(\varepsilon_i) \\
MSE^{AL} &= E[(b_i - \varepsilon_i)^2] \\
&= E(b_i^2) + var(\varepsilon_i) \\
MSE^{Com} &= E[((\beta - 1)p_i + \beta b_i - \varepsilon_i)^2] \\
&= (\beta - 1)^2 E(p^2) + \beta^2 E(b_i^2) + var(\varepsilon_i) - 2\beta(1 - \beta)E(p_i)E(b_i)
\end{aligned}$$

Assumption 2 *With a large forecasting sample size and applying the central limit theorem, we assume that the expected value of the private information component averages out across different companies, such that* $E(p) = E[\mathbb{E}(y_i|X_i, P_i) - \mathbb{E}(y_i|X_i)] = 0$.

With $E(p) = 0$ as specified in Assumption 2, the difference in MSE can be calculated as follows:

$$\begin{aligned} MSE^{Com} - MSE^{L} &= \beta(\beta-2)E(p_i^2) + \beta^2 E(b_i^2) - E(\eta_i^2) + 2\beta(\beta-1)E(p_i)E(b_i) + 2E(p_i)E(\eta_i) \\ &= \beta(\beta-2)E(p_i^2) + \beta^2 E(b_i^2) - E(\eta_i^2), \qquad \text{(i)} \\ MSE^{Com} - MSE^{AL} &= (1-\beta)^2 E(p_i^2) - (1-\beta^2)E(b_i^2) + 2\beta(\beta-1)E(p_i)E(b_i) \\ &= (1-\beta)^2 E(p_i^2) - (1-\beta^2)E(b_i^2). \qquad \text{(ii)} \end{aligned}$$

To determine the solution for the weight parameter β that ensures the first difference (i) is less than zero, indicating that the combined forecast outperforms Lintner's forecast, we can equivalently seek solutions for the inequality $\beta(\beta-2)E(p_i^2) + \beta^2 E(b_i^2) < 0$. Since the expectation of a squared term, $E(\eta_i^2)$, is non-negative, solutions to this simplified inequality satisfy the original inequality (i). Additionally, considering that β ranges from 0 to 1, the sufficient condition for making (i) negative is:

$$\beta < \frac{2E(p_i^2)}{E(p_i^2) + E(b_i^2)}.$$

The sufficient condition to make (ii) negative can also be determined as:

$$\beta > \frac{E(p_i^2) - E(b_i^2)}{E(p_i^2) + E(b_i^2)}.$$

Therefore, Lemma 4 is proven.

Appendix B. Details on machine learning forecast formation

Appendix B.1. Lists of input variables

Table B1 presents our data sources and the variables used to construct the machine learning forecast of DPS. We obtain quarterly financial ratios from WRDS through the CRSP/Compustat Merged quarterly fundamentals dataset. The calculation of financial ratios follows the method recommended by the Financial Ratios Suite available on WRDS. To ensure data availability at the forecast time points, all variables are based on one-quarter lagged values. We match the I/B/E/S actual file with the summary file containing analysts' consensus forecasts, using Ticker and forecast fiscal period as matching criteria. To facilitate the merging of I/B/E/S with other datasets, we use the link table provided by WRDS, ensuring that the firm's historical CUSIP matches in both I/B/E/S and CRSP datasets.

Table B1: Variable Definitions and Categories

Variable labels	**Definition**	**Category**
AL_h_DPS	analyst forecast of h-period-ahead DPS	Analysis forecasts
AL_h_EPS	analyst forecast of h-period-ahead EPS	Analysis forecasts
L_h_DPS	past period of actual DPS	Historical actual
L_h_EPS	past period of actual EPS	Historical actual
prc	monthly price	Stock information
ret	monthly total return	Stock information
shrout	monthly shares outstanding	Stock information
de_ratio	Total Debt/Equity	Solvency
debt_assets	Total Debt/Total Assets	Solvency
capital_ratio	Capitalization Ratio	Capitalization
totdebt_invcap	Total Debt/Invested Capital	Capitalization
equity_invcap	Common Equity/Invested Capital	Capitalization
debt_invcap	Long-term Debt/Invested Capital	Capitalization
sale_invcap	Sales/Invested Capital	Efficiency
at_turn	Asset Turnover	Efficiency
sale_equity	Sales/Stockholders Equity	Efficiency
lt_debt	Long-term Debt/Total Liabilities	Financial Soundness
cash_lt	Cash Balance/Total Liabilities	Financial Soundness
npm	Net Profit Margin	Profitability
opmad	Operating Profit Margin After Depreciation	Profitability

Table B1 continued from previous page

Variables labels	Definition	Category
aftret_equity	After-tax Return on Total Stockholders' Equity	Profitability
roa	Return on Assets	Profitability
roe	Return on Equity	Profitability
pe_op_basic	Price/Operating Earnings (Basic, Excl. EI)	Valuation
bm	book-to-market ratio	Valuation
ptb	Price/Book	Valuation
ps	Price/Sales	Valuation
retain_eps	retained earnings per share	Fundamentals
txtq	income taxes	Fundamentals
oancfy	Operating Activities - Net Cash Flow	Fundamentals
ivncfy	Investing Activities - Net Cash Flow	Fundamentals
fincfy	Financing Activities - Net Cash Flow	Fundamentals
chechy	Cash and Cash Equivalents Increase (Decrease)	Fundamentals
capxy	Capital Expenditures	Fundamentals
fopoy	Funds from Operations	Fundamentals
ppentq	Property Plant and Equipment	Fundamentals
txditcq	Deferred Taxes and Investment Tax Credit	Fundamentals
nopiq	Non-Operating Income (Expense)	Fundamentals
apq	Account Payable/Creditors	Fundamentals
xoprq	Operating Expense	Fundamentals
mibtq	Noncontrolling Interests	Fundamentals
capsq	Capital Surplus/Share Premium Reserve	Fundamentals
xsgaq	Selling, General and Administrative Expenses	Fundamentals
dpq	Depreciation and Amortization	Fundamentals
size	Capitalization	Fundamentals
niq	Net Income	Fundamentals
ibq	Income Before Extraordinary Items	Fundamentals
cstkeq	Common Stock Equivalents	Fundamentals
ivchy	Increase in Investments	Fundamentals
sivy	Sale of Investments	Fundamentals
dltisy	Long-Term Debt - Issuance	Fundamentals
dltry	Long-Term Debt - Reduction	Fundamentals
opepsy	Earnings Per Share from Operations	Fundamentals
aqcy	Acquisitions	Fundamentals
tstkq	Treasury Stock	Fundamentals
dlcq	Debt in Current Liabilities	Fundamentals
revtq	Revenue	Fundamentals
capsq	Capital Surplus/Share Premium Reserve	Fundamentals
cstkq	Common/Ordinary Stock (Capital)	Fundamentals
rectq	Receivables	Fundamentals
invtq	Inventories	Fundamentals
pstkrq	Preferred/Preference Stock	Fundamentals
atq	total asset	Fundamentals
seqq	Stockholders Equity	Fundamentals

Table B1 continued from previous page

Variables labels	Definition	Category
repurch	repurchases scaled by shares: (prstkc-pstkrv)/shares	Fundamentals
gdp	GDP gross	Macro economics
cg	consumption growth	Macro economics
indg	Growth of industrial production	Macro economics
unemp	Unemployment rate	Macro economics

Appendix B.2. Description of three tree-based techniques

We apply three ensemble tree-based regression methods: random forest (RF), gradient-boost (GB) trees, and extreme gradient-boosting (XGB) trees to make more accurate, robust and stable predictions.

In RF, the fundamental idea revolves around bootstrapping samples of the training dataset and constructing a decision tree for each of these samples. Specifically, given a training dataset D of size N, for each of the k trees to be grown, a sample of size N is drawn from D with replacement. For each decision tree split, a random subset of m predictors (features) is selected from the total p predictors, and the best split on these m is used to split the node. The typical choice for m is $\sqrt{p}$ for regression problems. The final prediction of the RF regressor is an average of the predictions from all the individual decision trees. Mathematically, for a new data point x, the RF regression prediction $\hat{y}$ can be expressed as: $\hat{y}(x) = \frac{1}{k}\sum_{i=1}^{k} h_i(x)$. Where $h_i(x)$ represents the prediction of the i^{th} decision tree for the data point x.

GB focuses on minimizing the residuals left by the previous trees. The algorithm begins by initializing with a simple model, often the mean or another straightforward statistic, and computes the residuals. For each subsequent step, a decision tree is trained to predict the negative gradient (or, essentially, the residuals) of the loss function with respect to the previous cumulative prediction. The output for this tree is then scaled by a factor known as the learning rate and added to the previous

predictions. Mathematically, given a loss function $L(y, F(x))$, where y is the true value and $F(x)$ is our predicted value, the negative gradient (or pseudo-residual) r at step m for every data point i can be expressed as: $.r_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}$. A tree is then fit to these residuals, and the resulting predictions are scaled by a factor ν (the learning rate) and added to the previous model: $F_m(x) = F_{m-1}(x) + \nu \sum_{j=1}^{J} \gamma_{jm} I(x \in R_{jm})$ where J is the number of terminal nodes, R_{jm} is the region of the j^{th} terminal node for the m^{th} tree, and γ_{jm} is the value predicted for region R_{jm}.

XGB is a sophisticated optimization of the traditional gradient boosting framework. Distinct from GB, XGB aims to minimize a regularized objective function:

$$Obj(\Theta) = \sum_i l(y_i, \hat{y}_i) + \sum_j \Omega(f_j)$$

where $l(y_i, \hat{y}_i)$ denotes the training loss, capturing the discrepancy between the predicted $\hat{y}_i$ and actual y_i values, while $\Omega(f_j)$ signifies a regularization term, penalizing model complexity to mitigate overfitting. This regularization integrates both L1 and L2 forms, offering a robust mechanism against potential overfitting. An important feature of XGB is its capability to handle missing data, intuitively imputing values during tree construction. Additionally, unlike traditional gradient boosting, which grows trees depth-first, XGB employs a depth-first strategy, subsequently pruning trees using the "max_depth" parameter, leading to optimized tree architectures. Integrated cross-validation during each boosting iteration facilitates pinpointing the optimal number of boosting rounds.

Appendix B.3. Forecast formation process for DPS

We construct our real-time forecasts for from one to three-quarters-ahead and one to three-years-ahead DPS using a rolling-window approach, split by time. This process involves six steps, starting from the original dataset:

1. The original dataset contains monthly firm-level values of predictor $X_{i,t}$, and includes realized future values of one and two-year-ahead DPS from 2003 to 2022.

2. We replace missing values in all variables, except for the realized DPS and the analyst forecast of DPS, with industry-time medians. Industries are classified according to the Fama-French 49 industry classification.

3. We create the first training dataset using the data from the year 2003. We winsorize all the variables at a 1% level and standardize all the predictors in the training set to have a mean of zero and a unit variance.

4. We set the year 2004 as the gap year for the one-year-ahead forecast and years 2004 and 2005 as the two gap years for the two-year-ahead forecast. We create the first test dataset using the data from the years 2005 (for the one-year-ahead forecast) and 2006 (for the two-year-ahead forecast). Similar to the training data, in the test data, we winsorize all the variables at a 1% level and standardize all the predictors using the mean and variance obtained from the training dataset

5. We fit the model using the training set data with one of the three tree-based methods. We apply a 5-fold cross-validation to tune the hyperparameters for each algorithm at first and then choose the top 10 most important features to fit the forecast model.

6. We use the fitted model to generate forecasts based on three algorithms on the test set. Moving time forward, we repeat Steps 3-5 until the last test set within the year of 2022.

We implement the RF and GB three regressions, along with their cross-validation program, using the `scikit-learn` package in Python. The XGB regression program is also supported by a dedicated

package `xgboost` in Python [1]. For the tree-based algorithms, we apply 5-fold cross-validation to select the optimal combination of hyperparameters from a specified range.

- **RF**. We employ a Grid Search methodology over a predefined parameter space supported by the `GridSearchCV` function. We set the number of trees, `n_estimators`, to be 1000. We choose the maximum depth of the trees, `max_depth`, between 4 and 8, controlling how deep each tree can grow and consequently its complexity. The minimum number of samples required to split an internal node, `min_samples_split`, is chosen from 2 and 8. We consider the proportion of features for the best split, `max_features`, from 0.5 (indicating half of the features) to 1 (using all features). The minimum number of samples required to be at a leaf node, `min_samples_leaf`, has values of 1 and 4.

- **GB**. We use the `GridSearchCV` function to choose parameters for the Gradient-Boost tree regression method. First, we select the number of trees, `n_estimators`, from 500 to 10,000. The maximum depth of the trees, `max_depth`, has possible values of 1 and 3. We consider the `learning_rate`, which determines the step size at each iteration while moving towards a minimum of the loss function, with values of 0.001 and 0.1.

- **XGB**. The `xgboost` program in Python supports the `GridSearchCV` function. Since `xgboost` allows the calibration of a large number of hyperparameters, we select critical ones following Chen et al. (2024) to avoid overcomplexity. We choose the number of trees, `n_estimators`, from 100 to 500. The maximum depth of the trees, `max_depth`, takes values of 4 and 8. `Learning_rate` is selected from 0.001 and 0.1. `min_child_weight`, selected from values 1 or 3, affects the sensitivity of the model to specific data patterns. The portion of data used for building each tree, `subsample`, is either 80% or 100%. Finally, for the data sampling method, `sampling_method`, we

[1] see https://xgboost.readthedocs.io/en/stable/python/python_intro.html for more information

experiment with both uniform and gradient-based approaches.

Appendix B.4. Computation hardware

For the tree-based regression methods, the rolling window prediction with 5-fold cross-validation is highly computationally intensive. To save computation time, we split the entire dataset by time and perform parallel computing using the cloud computing service provided by the Edinburgh Compute and Data Facility (ECDF [2]). Each estimation uses 20 cores, each with 4GB of RAM, resulting in a total of 40 CPUs, and takes around 8 hours.

Appendix C. The EPS-based expected return calculations

In this section, we describe how we calculate the conventional EPS-based expected return measures, following the approach used by Pástor et al. (2008), Lee et al. (2009), and Li et al. (2013). We denote the EPS-based expected return as r_e. We obtain EPS expectations, $E_t[EPS_{t+k}]$, up to three years ahead from either machine learning forecasts or analyst forecasts of EPS. If the machine learning forecast of EPS is unavailable or negative, we replace it with the available and positive analyst forecast values. To generate machine learning forecasts of EPS, we follow the approach outlined by van Binsbergen et al. (2022). This approach employs a random forest regression model that considers predictors such as analyst forecasts, firm financial ratios from WRDS, and macroeconomic variables. This machine learning forecast serves as a statistical benchmark expectation, helping to isolate substantial subjective bias in analyst forecasts of earnings.

Similar to our DPS forecast, the machine learning forecast of EPS is generated using a rolling window approach and is designed to be out-of-sample. We select hyperparameters for the model based on the values suggested by van Binsbergen et al. (2022): `n_estimators` is set to be 2000, `max_depth` is

[2] www.ecdf.ed.ac.uk

7, `min_samples_split` is 1%, and `min_samples_leaf` is 5.

For missing values and negative forecasts of EPS, we use the most recent available realized values, following the approach suggested by Li et al. (2013). For instance, if $E_t[EPS_{t+2}]$ is missing or negative and $E_t[EPS_{t+1}] > 0$, and the most recent realized $EPS_t > 0$, then $E_t[EPS_{t+2}]$ is replaced by $E_t[EPS_{t+1}] \times (E_t[EPS_{t+1}]/EPS_t)$. Similarly, the negative $E_t[EPS_{t+1}]$ is replaced by $EPS_t \times (1 + (E_t[EPS_{t+2}]/EPS_t - 1)^{1/2})$. If $E_t[EPS_{t+3}]$ is missing or negative, we use $E_t[EPS_{t+2}]$ times the analyst forecast of long-term EPS growth rate. Eventually, r_e is obtained by solving the following nonlinear finite Gordon growth model:

$$P_t = \sum_{k=1}^{T=15} \frac{E_t[EPS_{t+k}] \times (1 - b_{t+k})}{(1 + r_{e,t})^k} + \frac{E_t[EPS_{t+k}]}{r_{e,t}(1 + r_{e,t})^T},$$

where b_{t+k} is the estimated plowback rate. The terminal value term is calculated using the no-growth perpetuity theory for the long run, with the formula $TV_{t+T} = E_t[EPS_{t+T+1}]/r_{ICC}$ (Damodaran, 2012). Plowback rate estimates that fall outside the range [0, 1] are adjusted to 0 and 1 accordingly.

To generate EPS forecast for $t + 4$ to $t + T + 1$, we assume that the earnings growth rate for year $t + 3$, $g_3 = E_t[EPS_{t+3}]/E_t[EPS_{t+2}] - 1$, exponentially reverts to a long-run steady value represented by the long-run nominal gross domestic product (GDP) growth rate. In cases where the growth rate g_3 is negative, we replace it with the year $t + 2$ earnings growth rate, denoted as g_2. Consequently, for $k = 4, ..., T + 1$, we calculate the earnings growth and earnings expectations using the following iterative process:

$$g_{t+k} = g_{t+k-1} \times exp[log(g/g_3)/T],$$

$$E_t[EPS_{t+k}] = E_t[EPS_{t+k-1}] \times (1 + g_{t+k}).$$

The initial one-year ahead plowback rate, denoted as b_1, for year $t+1$ is estimated as one minus the most recent year's total payout ratio, which is available from the Financial Ratios Suite by WRDS. We assume that the plowback rates revert linearly to a steady-state plowback rate, denoted as b, following the formula: $b_{t+k} = b_{t+k-1} - (b_1 - b)/T$. According to the sustainable growth rate formula, in the steady state, the earnings growth rate equals the return on new investment (ROI) multiplied by the plowback rate. We further assume that ROI is the same as the expected return for new investments. Therefore, the plowback rates for years $t+3$ to $t+T$ can be computed as $b_{t+k} = b_{t+k-1} - (b_1 - b)/T$ for $k = 3, ..., T$.

Appendix D. Additional Predictive Regression Tests for expected return proxies)

In this section, we perform a robustness check on our expected return by assessing its ability to predict the aggregate market return represented by the CRSP index. We use expected return proxies aggregated across the entire available sample, allowing us to evaluate the effectiveness of all the expected return measures without the restriction of the S&P 500 stock universe. The three samples, each containing ICC values ($r_d, r_{e,AL}, r_{e,ML}$) within the range of 0 to 0.3 suggested by Hommel et al. (2023), span from January 2005 to December 2021 and consist of approximately 500,000 firm-month observations. We construct the aggregate ERP by either equal-weighted or value-weighted methods to predict the corresponding CRSP NYSE/NYSE American/NASDAQ/ARCA returns obtained from WRDS. Table D1 summarizes the results of the predictive regression and also the scaled measurement errors variance ($SMEV$).

In terms of the performance of r_d, its predictive power for one-month-ahead equal-weighted CRSP return is somewhat weaker compared to the equal-weighted S&P 500 return. However, it still stands out as the best performing expected return proxy among the alternatives, as also indicated by its most negative scaled measurement error variance. On the other hand, both r_e measures continue to exhibit

Table D1: Predictive regression for equal and value-weighted CRSP market return.
This table reports predictive regression results and the scaled measurement error variance in the prediction of CRSP market excess returns from January 2005 to December 2021. The left and right panels display results for equal-weighted and value-weighted returns, respectively. The first three columns display the estimated coefficient, Newey and West (1987) t-statistic with the number of lags equal to the forecast length, and R squared obtained from the predictive regression as defined above: $\sum_{h=1}^{H}(r_{t+h})/H = \beta_0 + \beta_1 \times M_t + \epsilon_{t+h}$.
where r_{t+h} represents h-periods-ahead compounded CRSP excess return. The vector of predictors M_t are defined the same as above tables. The fourth column $SMEV$ represents the scaled measurement error variance calculated by $SMEV_i = var_i(\hat{er}_{i,t}) - 2cov_i(r_{i,t+h}, \hat{er}_{i,t})$.

		Equal-weighted				Value-weighted			
variable	Horizon	β_1	t-stat	R-sq (%)	$SMEV$	β_1	t-stat	R-sq (%)	$SMEV$
r_d	1	2.736	1.603	3.85	-0.0063	3.989	2.332	5.93	-0.0069
	6	3.331	4.851	24.61	-0.0082	4.488	6.275	32.87	-0.0081
	12	2.783	7.929	36.14	-0.0068	3.444	5.668	41.13	-0.0062
$r_{e,AL}$	1	0.452	0.729	0.69	0.0009	0.297	0.484	0.34	0.0042
	6	0.721	1.769	7.60	-0.0042	0.462	1.027	3.59	0.0008
	12	0.748	3.563	16.80	-0.0047	0.524	2.201	9.33	-0.0005
$r_{e,ML}$	1	0.755	1.538	2.44	-0.0060	0.536	1.186	1.72	-0.0012
	6	1.078	4.268	21.44	-0.0139	0.698	2.559	12.73	-0.0065
	12	0.938	5.926	33.23	-0.0105	0.562	3.662	16.83	-0.0020
$Default$	1	0.909	0.574	0.63	-0.0017	1.087	0.677	0.92	-0.0024
	6	1.735	2.342	9.65	-0.0052	1.861	2.612	11.58	-0.0057
	12	1.637	6.316	17.75	-0.0048	1.685	7.044	19.79	-0.0050
$Term$	1	0.211	0.485	0.12	0.0043	0.110	0.257	0.03	0.0057
	6	0.183	0.547	0.39	0.0048	0.081	0.241	0.08	0.0063
	12	0.252	0.844	1.54	0.0039	0.146	0.480	0.55	0.0055
D/P	1	2.043	0.976	1.67	-0.0034	2.235	1.049	2.05	-0.0038
	6	2.735	2.673	11.64	-0.0045	2.817	2.743	12.86	-0.0047
	12	2.685	6.334	21.35	-0.0041	2.647	6.086	21.86	-0.0041
E/P	1	-0.317	-0.761	0.62	0.0275	-0.355	-0.838	0.79	0.0288
	6	-0.453	-1.447	5.39	0.0326	-0.482	-1.548	6.35	0.0335
	12	-0.417	-2.106	9.21	0.0309	-0.440	-2.325	10.80	0.0317
$payout$	1	0.040	1.458	2.52	4.0607	0.043	1.523	2.99	4.0347
	6	0.043	3.407	12.58	4.1045	0.044	3.410	13.99	4.0919
	12	0.034	4.684	16.79	4.3029	0.034	4.608	18.10	4.2992

limited predictive power for CRSP return across various horizons, with $r_{e,ML}$ showing relatively better performance compared to $r_{e,AL}$. The performance of other valuation ratios and business cycle variables remains consistent with previous findings.

Table D2 presents the results of the out-of-sample prediction tests for CRSP returns. r_d maintains

robust and significant predictive power for CRSP returns in all cases. Interestingly, the EPS-based expected return proxy, derived from the machine learning forecast of EPS, also demonstrates positive predictive performance with a 2.51% R^2_{OS} when using a 60-month expanding window estimation for the equal-weighted CRSP return. Additionally, the business cycle variable default spread exhibits significant and positive R^2_{OS} for equal-weighted CRSP returns.

Table D2: Out-of-sample predictive regression test for CRSP market return.
This table presents out-of-sample predictive test results for one-month-ahead CRSP excess returns, covering the period from January 2005 to December 2021. The forecast period spans from January 2010 (2015) for a 60 (120) months estimation window and ends at December 2021. The top and bottom panels display results for equal-weighted and value-weighted returns, respectively. The left and right panels indicate an estimation strategy with a rolling window and expanding window, respectively. Definitions of the tested ERP are the same as in the above tables. The first column reports out-of-sample R square R^2_{OS} calculated by: $R^2_{OS} = 1 - \frac{\sum_{k=1}^{T-m}(r_{m+k} - \hat{r}_{m+k})^2}{\sum_{k=1}^{T-m}(r_{m+k} - \bar{r}_{m+k})^2}$. The second column p-val stands for the p-value from the one-side test results using the adjusted-MSPE statistic regress on a constant, calculated as: $f_{t+1} = (r_{t+1} - \bar{r}_{t+1})^2 - [(r_{t+1} - \hat{r}_{t+1})^2 - (\bar{r}_{t+1} - \hat{r}_{t+1})^2]$. *, **, and *** indicate significance at 10%, 5%, and 1% levels, respectively.

Equal-weighted	Rolling window				Expanding window			
	60 months		120 months		60 months		120 months	
	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val
r_d	6.64**	0.029	12.35**	0.019	5.05**	0.010	6.80**	0.016
$r_{e,AL}$	−3.78		0.67	0.226	0.70	0.185	0.37	0.305
$r_{e,ML}$	−13.19		−1.36		2.51	0.099	1.52	0.217
$Default$	1.40**	0.048	3.45**	0.015	0.50	0.132	0.81	0.106
$Term$	−4.44		−1.62		−0.65		−0.85	
D/P	−2.03		−3.14		0.77	0.135	0.52	0.212
E/P	−7.72		−2.15		−1.75		0.55	0.291
$payout$	−0.65		2.60*	0.082	−0.08		2.38**	0.031
Value-weighted	Rolling window				Expanding window			
	60 months		120 months		60 months		120 months	
	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val	$R^2_{OS}(\%)$	p-val
r_d	2.81**	0.021	2.03*	0.068	3.42**	0.016	3.13*	0.059
$r_{e,AL}$	−1.24		−0.25		−0.39		−0.39	
$r_{e,ML}$	−5.16		−2.54		−0.24		−0.43	
$Default$	−2.82		−5.68		−0.03		−0.33	
$Term$	−5.63		−1.89		−0.74		−0.77	
D/P	−3.13		0.04	0.337	0.10	0.351	−0.20	
E/P	−0.36		2.26	0.201	−0.76		−0.58	
$payout$	−2.96		−1.86		−0.51		0.53	0.171

References

Chen, Y., Calabrese, R., and Martin-Barragan, B. (2024). Interpretable machine learning for imbalanced credit scoring datasets. *European Journal of Operational Research*, 312(1):357–372.

Damodaran, A. (2012). *Investment valuation: Tools and techniques for determining the value of any asset*, volume 666. John Wiley & Sons.

de Silva, T. and Thesmar, D. (2021). Noise in expectations: Evidence from analyst forecasts. Working Paper 28963, National Bureau of Economic Research.

Hommel, N., Landier, A., and Thesmar, D. (2023). Corporate valuation: An empirical comparison of discounting methods. Working Paper 30898, National Bureau of Economic Research.

Lee, C., Ng, D., and Swaminathan, B. (2009). Testing international asset pricing models using implied costs of capital. *The Journal of Financial and Quantitative Analysis*, 44(2):307–335.

Li, Y., Ng, D. T., and Swaminathan, B. (2013). Predicting market returns using aggregate implied cost of capital. *Journal of Financial Economics*, 110(2):419–436.

Newey, W. K. and West, K. D. (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica: Journal of the Econometric Society*, pages 703–708.

Pástor, L., Sinha, M., and Swaminathan, B. (2008). Estimating the intertemporal risk–return tradeoff using the implied cost of capital. *The Journal of Finance*, 63(6):2859–2897.

van Binsbergen, J. H., Han, X., and Lopez-Lira, A. (2022). Man versus Machine Learning: The Term Structure of Earnings Expectations and Conditional Biases. *The Review of Financial Studies*, 36(6):2361–2396.