Journal of Financial Economics

Growing the efficient frontier on panel trees[☆]

Lin William Cong [a,d,e], Guanhao Feng [b,*], Jingyu He [b], Xin He [c]

[a] SC Johnson College of Business, Cornell University, United States of America
[b] City University of Hong Kong, Hong Kong SAR, China
[c] International Institute of Finance, Faculty of Business for Science and Technology, School of Management, University of Science and Technology of China, China
[d] Fudan University FISF, China
[e] NBER, United States of America

ARTICLE INFO

JEL classification:
C1
G11
G12

Keywords:
Decision tree
Factors
Generative models
Interpretable AI
Test assets

ABSTRACT

We introduce a new class of tree-based models, P-Trees, for analyzing (unbalanced) panel of individual asset returns, generalizing high-dimensional sorting with economic guidance and interpretability. Under the mean–variance efficient framework, P-Trees construct test assets that significantly advance the efficient frontier compared to commonly used test assets, with alphas unexplained by benchmark pricing models. P-Tree tangency portfolios also constitute traded factors, recovering the pricing kernel and outperforming popular observable and latent factor models for investments and cross-sectional pricing. Finally, P-Trees capture the complexity of asset returns with sparsity, achieving out-of-sample Sharpe ratios close to those attained only by over-parameterized large models.

1. Introduction

Estimating the mean–variance efficient (MVE) frontier is crucial for asset pricing and investment management. Yet, estimating the tangency portfolio (Markowitz, 1952) using the unbalanced panel of thousands of individual asset returns proves impracticable. Empirical studies typically consider a "diversified" set of test assets (e.g., ME-BM 25 portfolios) to estimate and evaluate factor models, hoping these test assets or a few common factors can span the same efficient frontier as individual assets. However, popular factor models hardly explain the cross section of conventional pre-specified test assets (e.g., Kozak et al., 2018; Lopez-Lira and Roussanov, 2020), not to mention

[☆] We are grateful to Fahiz Baba-Yara (discussant), Svetlana Bryzgalova (discussant), John Campbell, Vic Chien, Andrei Goncalves, Bing Han, Kewei Hou, Bob Jarrow, Ron Kaniel, Chris Malloy, Alberto Rossi (discussant), Gideon Saar, Shrihari Santosh (discussant), Artem Streltsov, Ziying Sun (discussant), Peixuan Yuan (discussant), and Guofu Zhou for detailed comments and suggestions. We also thank Doron Avramov, Jie Cao, Zhanhui Chen, Eric Ghysels, P. Richard Hahn, Dashan Huang, Sophia Zhengzi Li, Asaf Manela, Stefan Nagel, Adam Reed, Artem Streltsov, Yinan Su, Fred Sun, Yaki Tsaig, Mao Ye, Dacheng Xiu, Bobby Yu, and seminar and conference participants at ArrowStreet Capital, 2024 AsianFA Annual Meeting, Baylor University, BUAA, Ca' Foscari University of Venice, Cambridge University Algorithmic Trading Society Quant Conference, CFTRC 2022, CityUHK, CKGSB, Conference on FinTech, Innovation and Development at Hangzhou (2nd), Cornell University, 2024 EFMA Annual Meeting, Goethe University Frankfurt, GSU-RFS FinTech Conference 2022, HKAIFT-Columbia joint seminar, Hunan University, INFORMS Annual Meeting 2021, 4th International FinTech Research Forum (RUC), 2023 Mid-South DATA Conference (Memphis), KAIST Digital Finance Conference, NFA 2022, University of Oxford SML-Fin Seminar, OSU, PHBS, PKU Guanghua, PKU-NSD, PKU-NUS Annual International Conference on Quantitative Finance and Economics, Qube Research and Technology, Reichman University (IDC Herzliya), Schroders Investments, Shanghai Financial Forefront Symposium (3rd), SHUFE, NUS, SMU, SUSTech, TAMU, Tsinghua SEM, UNC, University of Bath, University of Hawai'i, USC, USTC, University of Macau, World Online Seminars on Machine Learning in Finance, WFA 2022, and 2023 XJTLU AI and Big Data in Accounting and Finance Conference for constructive discussions and feedback. Feng's research is partly supported by the Hong Kong Research Grants Council (GRF-11502721, GRF-11502023) and the National Natural Science Foundation of China (NSFC-72203190). He J.'s research is partly supported by the Hong Kong Research Grants Council (ECS-21504921, GRF-11507022, GRF-11509224) and the National Natural Science Foundation of China (NSFC-72403214). Feng and He J. are partly supported by the InnoHK initiative and the Laboratory for AI-Powered Financial Technologies. The authors also acknowledge research awards from INQUIRE Europe and IQAM Research.

Received 20 November 2023; Received in revised form 4 February 2025; Accepted 5 February 2025

Available online 18 February 2025

the ad hoc nature of these test assets hampers the effectiveness of model estimations and evaluations (Lewellen et al., 2010; Ang et al., 2020). For example, characteristics-based test assets are often limited to univariate- and bivariate-sorted portfolios due to the challenges of high-dimensional sorting (Cochrane, 2011), overlooking nonlinearity and asymmetric interactions (that do not uniformly apply to all assets), even with dependent sorting (Daniel et al., 1997). These problems cannot be fully addressed by evaluating different test assets separately for robustness checks (e.g., Fama and French, 1996), selecting test assets (Daniel et al., 2020; Giglio et al., 2023; Bryzgalova et al., 2023), or combining test assets (e.g., Feng et al., 2020).

The evaluation of factor models and test assets is about mean–variance diversification and, therefore, the performance of the tangency portfolio constructed using these test assets or factors. Cochrane (2011) states that expected returns, variances, and covariances are stable functions of asset characteristics (e.g., size, value), and Kelly et al. (2019) further show return-factor covariances are associated with characteristics. The key to bridging the gap between the ultimate efficient frontier and tangency portfolios of test assets or factors spanning the SDF therefore lies in systematically utilizing the high-dimensional asset characteristics, which contain rich information on the joint distribution of asset returns dynamics (e.g., Kozak and Nagel, 2022).

To achieve this, we propose a new approach that integrates nonlinearity and asymmetric interactions with the high-dimensional characteristics to create test assets that span the efficient frontier of individual asset returns. The Panel Tree (P-Tree) framework is inspired by modern AI and constitutes a versatile family of models that cluster panel observations to achieve given economic objectives. Creating test assets requires clustering individual assets into groups to form portfolios.[1] Under a global split criterion for goal-oriented search, P-Tree clusters individual asset returns and creates basis portfolios. It utilizes high-dimensional characteristics under the MVE framework to jointly generate test assets and a latent factor, which is the MVE portfolio of test assets, and recovers the stochastic discount factor (SDF). Therefore, P-Tree extends the scope of regression tree models beyond pure prediction tasks, particularly generating diversified test assets that reach the ultimate efficient frontier.[2]

Specifically, P-Tree employs a “top-down” approach (typical trees are drawn with the root at the top), splitting the cross section of thousands of individual assets and grouping them into a small number of clusters based on characteristic values to form (value-weighted) portfolios. Guided by asset pricing considerations, we grow a P-Tree to iteratively construct test assets and latent factors for the SDF, following the baseline specification of the global split criterion that maximizes the Sharpe ratio of the SDF. The high Sharpe ratio of the constructed SDF reflects the high MVE frontier spanned by the generated leaf basis portfolios. The resulting leaf basis portfolios and latent factors provide researchers with a diversified set of test assets and the SDF model. Furthermore, P-Trees are intuitive and transparent, allowing economic interpretation and identifying a sparse set of useful characteristics that interact to generate test assets and latent factors jointly.

Fundamentally, P-Tree is a greedy search algorithm guided by asset pricing goals for optimal clustering based on similar characteristic values within a large modeling space. In empirical asset pricing, basis portfolios are usually formed by grouping assets based on classifications such as countries or industries, or through unsupervised clustering (e.g., security sorting) based on return correlations or characteristics. In contrast, P-Tree generates characteristics-managed leaf basis portfolios to maximize the Sharpe ratio of their MVE portfolios, without being either supervised or unsupervised. The objective can be flexibly specified, making the P-Tree framework applicable for a wide range of panel data analyses.[3]

Methodological innovations. Our asset pricing application effectively demonstrates the key methodological innovations. First, off-the-shelf ML methods, including the famous Classification and Regression Trees (CART, Breiman et al., 1984), typically assume that observations are i.i.d., and not designed for analyzing panel data. Though naively fitting CART or tree ensembles (boosted trees or random forest) on characteristics-return data shows positive predictive performance (e.g., Gu et al., 2020), they ignore the panel data structure. Alternatively, P-Tree fully embraces the panel data structure and incorporates a time-invariant tree structure for multi-period observations. The time-invariant P-Tree allows for economic interpretability when building ML models on panel data. For example, similar to security sorting, the time-invariant P-Tree allows the same set of characteristics-managed leaf basis portfolios for all periods.

Second, the standard tree-based models in ML, including CART, focus purely on prediction. Furthermore, CART-based models grow recursively, optimizing the quadratic loss of prediction. These models optimize locally at each node without considering sibling nodes, mainly for computational efficiency. However, this “myopic” strategy often leads to overfitting, because it operates on fewer observations in each node as the tree grows. By contrast, P-Trees broaden the applications beyond prediction, encompassing the generation of test assets and latent factors. The iterative growth of P-Trees is designed to utilize data from the entire cross section to guard against the overfitting that afflicts conventional trees grown using local split criteria. P-Trees combine economic principles and nonlinear ML algorithms while ensuring the interpretability of the graphical tree diagram, providing a unified approach for splitting the cross section and growing the efficient frontier.

Third, the P-Tree framework can integrate the boosting or bagging strategy of ML, enabling multiple P-Trees to form a multi-factor model. On the one hand, the Boosted P-Tree grows additional P-Trees based on the previous ones, providing a unified framework for a multi-factor model, such that additional factors and test assets must provide an incremental contribution. The ensemble approach makes P-Tree a versatile tool that can enhance the performance of any predetermined factor model by exploiting the unspanned efficient frontier under the MVE framework. When boosted to generate multiple factors, P-Trees offer an alternative to principal component analysis (PCA) and deep neural networks, with greater interpretability and sparsity, while capturing (asymmetric) interactions.[4] On the other hand, the random P-Forest generates additional P-Trees on random bootstrap samples, offering a unified framework to create multiple sets of uncorrelated factors and test assets, which can be used to assess characteristic importance. A large random P-Forest is also connected to the recent literature of large and over-parameterized models (e.g., Didisheim et al., 2024).

Empirical findings. First, we study monthly U.S. equity returns from 1981 to 2020 using 61 firm characteristics. The P-Tree tangency portfolios are constructed as traded factors, significantly advancing the efficient frontier with high annualized Sharpe ratios, ranging from 6.37 for a single P-Tree (with 10 portfolios) to 15.63 for 20 boosted P-Trees (with 200 portfolios). These numbers are substantially higher

[1] The commonly used security sorting in empirical research is one type of unsupervised clustering based on firm characteristics, similar to the decision tree (Bryzgalova et al., 2023). However, no unified method has been developed for sorting securities and generating test assets.

[2] Tree-based models excel in predicting complex data with high dimensionality, nonlinearity, and variable interactions, even in low signal-to-noise environments and small sample sizes (e.g., Sorensen et al., 2000; Rossi and Timmermann, 2015; Gu et al., 2020; Bianchi et al., 2021; Capponi and Yu, 2024).

[3] We have developed and shared the P-Tree package, `PTree`, in a public repository in R for other researchers to explore (see https://github.com/Quantactix/PTree).

[4] Other recent studies explore latent factors on PCA (Kelly et al., 2019; Lettau and Pelger, 2020; Kim et al., 2021; He et al., 2023) and deep learning (e.g., Gu et al., 2021; Chen et al., 2024; Feng et al., 2024).

than those constructed by conventional basis portfolios (univariate- or bivariate-sorted portfolios) or commonly used factor models. These findings provide strong evidence of a significant gap between the current empirical literature and the potential limits of the efficient frontier. Moreover, generated under the unified MVE framework, boosted P-Trees produce multi-factor models that effectively price the cross-sectional returns. Also, P-Tree factors offer annualized Sharpe ratios over 3 and significantly positive alphas in OOS analyses for past-predicting-future and future-predicting-past tests.[5]

Second, these diversified P-Tree test assets span the efficient frontier and pose a significant challenge for alternative factor models, highlighting the importance of test assets. We identify many economically and statistically unexplained test assets, indicated by most monthly alphas larger than 1%, and an extremely high GRS test statistic of 141.27 (aggregated weighted pricing error) for the first P-Tree against the Fama–French five-factor model (Fama and French, 2015). These asymmetric, interactive, and nonlinear characteristics-managed leaf basis portfolios are difficult to explain using other well-known observable factor models (e.g., Hou et al., 2021) and ML-based latent factor models (e.g., Kelly et al., 2019; Lettau and Pelger, 2020). Given the insufficient MVE spanning and low hurdles of univariate- or bivariate-sorted portfolios for model testing, we recommend including these multi-characteristic, systematically sorted P-Tree test assets in future model evaluations.

Finally, we design a random P-Forest with bagging to evaluate characteristic importance, account for variable selection risk, and validate transparent and interpretable P-Tree models. We confirm that the same small set of characteristics (e.g., SUE, DOLVOL, and BM_IA) selected by P-Trees are likely proxies for the true fundamental risk inherent in the SDF, which may be overlooked in a linear factor model with ad hoc selected factors. In addition, the random P-Forest SDF as a large regularized model has an excellent OOS Sharpe ratio, which shows similar patterns to the random split SDF but is significantly more efficient in computation. In a sense, goal-oriented search improves upon brute-force large models with statistical and economic regularizations, effectively combining economic objectives with ML models. Finally, P-Trees display all split rules, aiding researchers in understanding feature interactions.[6]

Literature. P-Trees constitute the first goal-oriented, systematic clustering of individual securities and generation of leaf basis portfolios under the MVE framework. Hoberg and Welch (2009) propose constructing factors and test assets by optimizing objective functions instead of sorting. For unsupervised clustering, Ahn et al. (2009) model within- and between-group return correlations to form portfolio groups, Chen et al. (2017) evaluate the performance by grouping hedge fund alpha values, and Patton and Weller (2022) group assets based on their heterogeneous risk premia on risk factors. We join Cong et al. (2023) as the earliest studies that iteratively cluster panel data by maximizing specific economic objectives.

Our latent factor model is related to but different from the recent regularized portfolio (or SDF) literature (e.g., Ao et al., 2019; Kozak et al., 2020; Bryzgalova et al., 2023), which typically estimate their regularized portfolio (or SDF) on a large number of prespecified test assets. In contrast, the SDF and test assets are generated iteratively in our unified P-Tree framework by maximizing the Sharpe ratio. Notably, Bryzgalova et al. (2023) shrink useless assets when estimating the optimal portfolio and describe this process as pruning or regularizing a decision tree (instead of growing it) working from potential leaf portfolios. They highlight the resemblance between the decision tree and security sorting but do not specify a split criterion for tree growth. Their paper manually specifies a small set of split candidates and a shallow depth for initial trees. P-Trees differ by growing the tree from the root to provide a goal-oriented clustering approach and efficiently scanning the large space of generalized sequential sorting on high-dimensional characteristics.

Moreover, our study contributes to the growing literature on latent factor models in asset pricing. For recent developments of PCA, in addition to the projected PCA (Kim et al., 2021) and reduced-rank approach (He et al., 2023), the instrumental PCA (IPCA, Kelly et al., 2019) uses characteristics to model the time-varying factor loadings and estimate principal components, and the risk premia PCA (RPPCA, Lettau and Pelger, 2020) adds asset pricing regularization in the objective. Recent studies, such as the auto-encoder (Gu et al., 2021), generative adversarial network (Chen et al., 2024), characteristics-sorted factor approximation (Feng et al., 2023, 2024), and structural neural network (Fan et al., 2022), have also developed customized deep learning models for nonlinear latent factor modeling. Following these developments, P-Trees provide a graphical representation for variable nonlinearity and asymmetric interactions, which PCA or deep learning methods do not offer.

While ML and tree-based methods are widely adopted in finance due to their powerful nonlinear modeling capabilities using high-dimensional characteristics (e.g., Gu et al., 2020; Bianchi et al., 2021; Bali et al., 2023), most studies either apply off-the-shelf ML models for prediction without considering the panel data structure or incorporating economic guidance, or focus on supervised or unsupervised learning and prediction tasks. We add to several recent exceptions involving generative models, including (Cong et al., 2023), which combines Bayesian spike-and-slab priors and Panel Trees for asset pricing with uncommon factors, Cong et al. (2021), which develops the first "large" model in finance for portfolio management using reinforcement learning, Creal and Kim (2023), which adapts Bayesian trees to currency returns, and Cong et al. (2024), which studies heterogeneity in return predictability and links that to trading profitability of a predictability-based anomaly.

The P-Tree framework has a notable extension, the random P-Forest, which connects to recent discussions on "benign overfitting" and "high-dimensional interpolation" (e.g., Belkin et al., 2019; Hastie et al., 2022) in statistics, as well as the corresponding "virtue of complexity" in financial contexts (e.g., Kelly et al., 2022, 2024; Didisheim et al., 2024). We corroborate these studies by showing that large tree-based models for which the number of parameters exceeds the number of observations perform well OOS, provided that appropriate statistical regularization is applied. Moreover, we contribute to recent research that supports incorporating economic restrictions when estimating and evaluating machine learning or factor models (e.g., Gagliardini et al., 2016; Avramov et al., 2023; Jensen et al., 2024).

Finally, our tree-based greedy algorithm demonstrates human-like intelligence through a "divide-and-conquer" strategy, offering a sparse, interpretable, and computationally efficient modeling alternative to modern AI, distinct from deep reinforcement learning.

The remainder of the article is as follows: Section 2 introduces the P-Tree models. Section 3 illustrates the empirical applications of a single P-Tree to split the cross-section and generate test assets. Section 4 demonstrates the empirical results of the boosted and multi-factor P-Trees. Section 5 discusses the random P-Forest and P-Tree's robustness to macro regimes. Section 6 concludes, and the internet appendix includes simulation and additional empirical discussions.

[5] There is a gap between the in- and out-of-sample Sharpe ratios. The paragraph "limits to learning" in Internet Appendix I, "Simulation", provides more discussion on this gap.

[6] P-Trees allow the long and short legs of a long-short portfolio to interact with different characteristics, thereby loading the portfolio on different leaf basis portfolios. This asymmetric interactive sorting contrasts with the traditional treatment of a long-short portfolio as a single asset and complements the pioneering work of Jarrow et al. (2023) in modeling the two legs of anomaly portfolios separately.

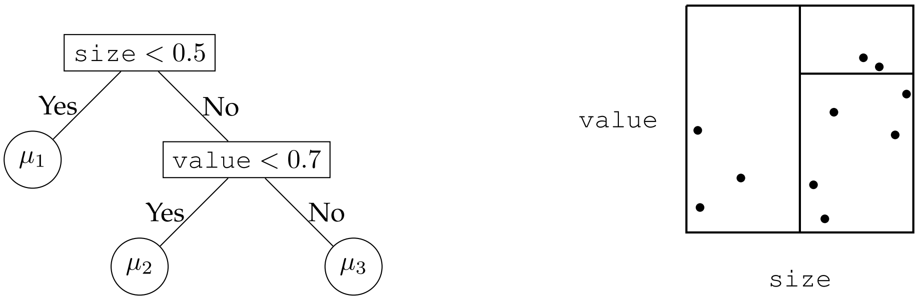


Fig. 1. Example of a decision tree.
Left: A decision tree has two splits, three leaf nodes, and three leaf parameters. Right: The corresponding partition plot is generated for the sample of predictor space on value and size.

2. Panel tree for asset pricing

Section 2.1 describes how P-Tree innovates over standard tree-based models. Section 2.2 delves into the growth of P-Trees, and Section 2.3 introduces boosted P-Trees for multi-factor models. Section 2.4 presents simulation results.

2.1. CART and P-Tree innovations

Designed for predictions, the standard CART (Breiman et al., 1984) model and its variants partition the predictor space into distinct, non-overlapping regions of leaf nodes and assign a constant leaf parameter to each region.[7] It searches and records the partition as follows: Suppose the data constitute K predictors, and the ith observation is denoted by $\mathbf{z}_i = (z_{i,1}, \dots, z_{i,K})$. The jth split rule of the decision tree is denoted by $\tilde{\mathbf{c}}^{(j)} = (z_{\cdot,k}, c_j)$, which partitions data by checking whether the value of the kth predictor $z_{\cdot,k}$ is greater or smaller than cutoff c_j. CART considers only binary splits, since any multiway-split tree can be represented as one with multiple binary splits. The optimal split rule is chosen to minimize the prediction error.

After J splits, CART partitions the predictor space into $J + 1$ leaf nodes denoted by $\{\mathcal{R}_n\}_{n=1}^{J+1}$ and assigns a constant leaf parameter μ_n to each node. The regression tree $\mathcal{T}$, with parameters $\Theta_J = \{\{\tilde{\mathbf{c}}^{(j)}\}_{j=1}^{J}, \{\mu_n\}_{n=1}^{J+1}\}$, constitutes a high-dimensional step function:

$$\mathcal{T}(\mathbf{z}_i \mid \Theta_J) = \sum_{n=1}^{J+1} \mu_n \mathbb{I}\left\{\mathbf{z}_i \in \mathcal{R}_j\right\}. \tag{1}$$

The indicator $\mathbb{I}\{\mathbf{z}_i \in \mathcal{R}_j\}$ takes 1 for one leaf node and 0 for others. The leaf parameters in the step function are estimated by averaging the training data within each leaf node. This nonparametric approach adapts CART to low signal-to-noise environments and small sample sizes. The tree model predicts new observations by locating the corresponding leaf and using its parameter as the prediction after training.

Fig. 1 illustrates how conventional tree-based models (e.g., CART) are applied to predict stock returns with firm characteristics, through recursively partitioning the characteristic space into clusters (small-cap, large-cap growth, and large-cap value). The bottom nodes, often called leaf nodes, are associated with constant leaf parameters, meaning assets within the same node share the same return forecast.

However, due to multiple limitations, these methods are unsuitable for economic or financial panel data and for generating test assets or pricing kernels. First, they treat the panel of asset returns as i.i.d observations. Second, they focus on predictions using statistical local split criteria, such as minimizing the sum of squared error objectives in the child nodes of a particular split. Thus, they do not incorporate panel structure or economic restrictions. As the tree grows, the number of observations in each node decreases, and idiosyncratic noise causes overfitting. While tree ensembles help mitigate overfitting, they are less interpretable than a single decision tree.

P-Tree addresses both issues by (i) utilizing a time-invariant tree structure and (ii) employing economic objectives that consider all observations, rather than just those in a parent node, to guide the tree growth. This (i) enables thorough extraction of panel data to construct leaf basis portfolios and (ii) iteratively builds P-Trees using global split criteria to prevent overfitting, extending the scope of trees from pure prediction to goal-oriented clustering for test asset and factor generation.

2.2. Growing a P-Tree

P-Tree partitions the universe of individual assets into non-overlapping leaf nodes based on the values of ranked characteristics with a time-invariant structure over T periods. The leaf basis portfolios are (value-weighted) portfolios of stocks within each leaf node from the time-invariant tree structure. P-Tree splits sequentially from the root node and generates one extra leaf basis portfolio after each additional split. Specifically, it produces $j + 1$ leaf basis portfolios after the jth split, reducing the dimension from thousands of individual assets to $j + 1$ portfolios. These characteristics-managed leaf basis portfolios expand on characteristics-sorted portfolios to accommodate asymmetric interactions of multiple characteristics.

Let $\mathbf{R}_t^{(j)}$ denote all excess return vectors of leaf basis portfolios after the jth split of the tree, and let $f_t^{(j)}$ denote the factor spanned by $\mathbf{R}_t^{(j)}$. The tree has $j + 1$ basis portfolios in all leaf nodes, denoted by $\mathbf{R}_t^{(j)} = [R_{1,t}^{(j)}, \dots, R_{n,t}^{(j)}, \dots, R_{j+1,t}^{(j)}]$, where $R_{n,t}^{(j)}$ represents a length-T vector of returns over T periods for the nth leaf node. To find the first split, we begin with a portfolio of all assets, denoted as $\mathbf{R}_t^{(0)} = \left[R_t^{(0)}\right]$, which serves as the root node for P-Tree. The tree is expanded through iterative updates of $\left\{\mathbf{R}_t^{(j)}, f_t^{(j)}\right\}$. First, the leaf basis portfolios $\mathbf{R}_t^{(j)}$ are expanded as the tree grows. Second, using the expanded leaf basis portfolios, we re-estimate the P-Tree factor $f_t^{(j)}$. Being a greedy algorithm, the sequential growth of the tree ensures computational feasibility compared to a full enumeration of possible sortings.

First split. The root node contains all assets corresponding to the value-weighted market factor. Firm characteristics are uniformly normalized cross-sectionally to the range of $[-1, 1]$ within each period. We assess different split threshold options c_m for characteristics $z_{\cdot,k}$, such as quintile splits $c_m \in [-0.6, -0.2, 0.2, 0.6]$.[8] Consider a candidate split

[7] CART is a binary decision tree model and serves as the foundation for ensemble methods such as random forest (Breiman, 2001) and boosting trees (Freund and Schapire, 1997). Other notable Bayesian tree models include BART (Chipman et al., 2010) and XBART (He et al., 2019; He and Hahn, 2023).

[8] Quintile splitting efficiently reduces the search space with only about four thousand stocks and many highly correlated characteristics for growing a P-Tree with 10 leaves. Decile or denser splitting may lead to overfitting and non-diversified test assets due to some portfolios having few stocks.

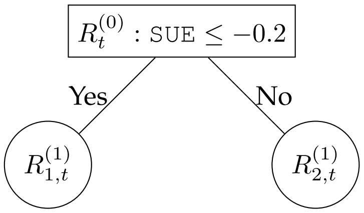


Fig. 2. Illustration of the first split.
To determine the optimal characteristic and cutpoint value, we evaluate a list of candidate splits (i.e., standardized unexpected quarterly earnings (SUE) ≤ −0.2; 40% percentile) for calculating the split criterion value.

$\tilde{\mathbf{c}}_{k,m} = (z_{\cdot,k}, c_m)$ as in Fig. 2. The candidate partitions the root node into left and right child nodes based on whether $z_{\cdot,k}$ is less than or equal to c_m.

The observations of stock returns within each potential child leaf form a leaf basis portfolio, with vectors of return $R_{1,t}^{(1)}$ and $R_{2,t}^{(1)}$, respectively. Note they are vectors of returns, thus maintaining the panel structure of asset returns. The P-Tree factor is estimated as the MVE portfolio of all leaf basis portfolios,

$$f_t^{(1)} = \mathbf{w}^{(1)\prime}\mathbf{R}_t^{(1)}, \quad \mathbf{w}^{(1)} \propto \hat{E}[\mathbf{R}_t^{(1)}\mathbf{R}_t^{(1)\prime}]^{-1}\hat{E}[\mathbf{R}_t^{(1)}], \tag{2}$$

where $\mathbf{R}_t^{(1)} = \left[R_{1,t}^{(1)}, R_{2,t}^{(1)}\right]$ is the matrix of returns of two leaf basis portfolios after the first split, $\hat{E}[\mathbf{R}^{(1)}]$ and $\hat{E}[\mathbf{R}_t^{(1)}\mathbf{R}_t^{(1)\prime}]$ are the sample mean and second moment matrix of leaf basis portfolios.[9] Generating latent factors aligns with the P-Tree split criteria, as the P-Tree factor represents the MVE portfolio of all leaf basis portfolios.

In the baseline specification, our split criterion therefore aims to maximize the Sharpe ratio of the MVE portfolio:

$$\mathcal{L}(\tilde{\mathbf{c}}_{k,m}) = \sqrt{\hat{\boldsymbol{\mu}}_{\mathbf{F}}' \hat{\boldsymbol{\Sigma}}_{\mathbf{F}}^{-1} \hat{\boldsymbol{\mu}}_{\mathbf{F}}}, \tag{4}$$

where $\mathbf{F} = f_t^{(1)}$ is the generated latent factor, with the sample mean $\hat{\boldsymbol{\mu}}_F$ and covariance matrix $\hat{\boldsymbol{\Sigma}}_{\mathbf{F}}$. The criterion allows P-Tree to construct and estimate basis portfolios, latent factors, and efficient portfolio weights simultaneously, for the global objective of constructing the efficient frontier.

Each candidate split $\tilde{\mathbf{c}}_{k,m}$ generates a different partition of the data, resulting in unique leaf basis portfolios, corresponding P-Tree factors, and ultimately, varying valuations of the split criteria in (4). Consequently, we loop over all candidate splits and select the one that maximizes the split criteria as our initial split rule.

Second split. The second split can occur at the root's left or right child node. We assess the split criteria for all candidate splits for *both* leaf nodes and choose the split that maximize the criteria as in (4). Fig. 3 depicts the tree of the candidates for the second split. In either scenario, one leaf node splits, becoming an internal node and generating two new leaf nodes. The P-Tree factor is then constructed based on *all three* leaf basis portfolios (thus global):

$$f_t^{(2)} = \mathbf{w}^{(2)\prime}\mathbf{R}_t^{(2)}, \quad \mathbf{w}^{(2)} \propto \hat{E}[\mathbf{R}_t^{(2)}\mathbf{R}_t^{(2)\prime}]^{-1}\hat{E}[\mathbf{R}_t^{(2)}], \tag{5}$$

where $\hat{E}[\mathbf{R}^{(2)}]$ and $\hat{E}[\mathbf{R}_t^{(2)}\mathbf{R}_t^{(2)\prime}]$ are the sample mean and second moment matrix of leaf basis portfolios $\mathbf{R}_t^{(2)} = \left[R_{1,t}^{(2)}, R_{2,t}^{(2)}, R_{3,t}^{(2)}\right]$ after the second split. The construction of the three basis portfolios depends on which node the candidate splits, as shown in Fig. 3. The updated P-Tree factor is plugged in to maximize the Sharpe ratio in (4), where $\mathbf{F} = f_t^{(2)}$ is the updated P-Tree factor.

Notably, our proposed objective is a *global* split criterion, because it considers all leaf basis portfolios when constructing $\mathbf{F}$ (the global MVE portfolio). Unlike CART, which focuses on a specific leaf node, our model explores all candidate splits in all leaf nodes to find the one with the largest investment improvement.

Growth termination. All subsequent splits proceed similarly. The tree-growing procedure is outlined in Algorithm 1. Determining the number of leaves is a natural turning point in tree growth, and it is also the only tuning parameter needed for P-Tree. We consider P-Trees with $J + 1 = 10$ leaf nodes in the baseline specification. Furthermore, we mandate a minimum leaf size of 20 for growing the tree because the leaf needs to serve as a basis portfolio, and any leaves that do not meet this criterion are not subjected to further splitting. Once the tree growing process terminates, it outputs the P-Tree split sequence, leaf basis portfolios $\mathbf{R}_t^{(J)} = [R_{1,t}^{(J)}, \ldots, R_{J+1,t}^{(J)}]$, and the P-Tree factor $f_t^{(J)}$. Note each leaf portfolio can be interpreted through economic fundamentals revealed by the sequential splits.

Possible extensions. The P-Tree framework is flexible to accommodate alternative objectives to the global split criteria in (4) for estimating the MVE portfolio using individual asset returns. For example, one might aim to create the minimum-variance portfolio or develop latent factors explaining test assets, such as individual or basis portfolio returns. The crucial aspect is defining a clear economic goal and utilizing the greedy growth algorithm within trees to explore the extensive observation clustering space and optimize the objective. P-Trees also allow for flexible greediness, enabling the consideration of multiple splits in each iteration instead of the conventional single split, or simultaneous splits along multiple characteristics by partitioning along some linear combinations of the characteristics. We leave these for future exploration.

2.3. Boosted P-Trees

Boosting is an ML technique that combines weak learners to form a strong learner (e.g., Freund and Schapire, 1997; Rossi and Timmermann, 2015; Capponi and Yu, 2024). Boosted P-Trees sequentially grow a list of additive trees that augment previous trees under the MVE framework, which helps further span the efficient frontier with additional basis portfolios and generate multi-factor models. Considering existing factors, the boosting procedure seamlessly integrates into the P-Tree framework by creating additional P-Tree factors to maximize the collective Sharpe ratio of the tangency portfolio. The boosting procedure is outlined below.

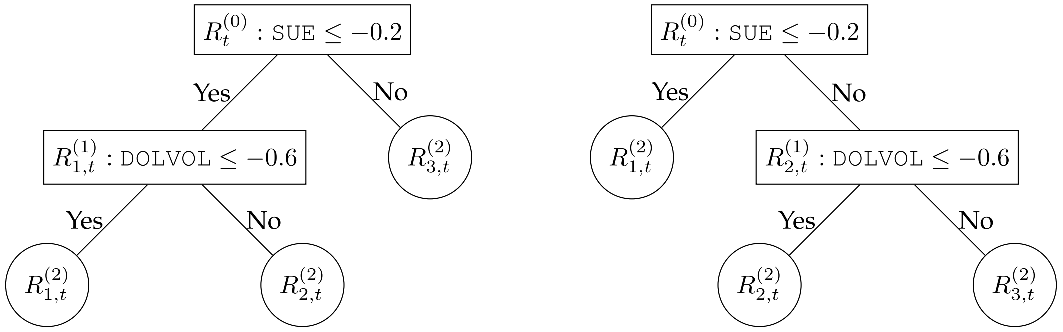
1. The initial factor, denoted as $f_{1,t}$, can be either generated by the first single P-Tree or benchmark factors selected by researchers (e.g., the market factor).
2. The second factor $f_{2,t}$ is generated by the second P-Tree to complement the first factor. The second tree growth follows the split criterion in (4), but $\mathbf{F} = [f_{1,t}, f_{2,t}]$. The MVE portfolio is generated directly from all the factors.[10]
3. Repeat Step 2 to sequentially generate P factors $\mathbf{F}_t = [f_{1,t}, \ldots, f_{P,t}]$ until the stopping criteria are triggered.

[9] Portfolio weights $\mathbf{w}^{(j)}$ are normalized such that the sum of absolute weights equals one. Following the regularization approaches in Kozak et al. (2020), Bryzgalova et al. (2023), Didisheim et al. (2024), we utilize small shrinkage parameters, $\gamma = 10^{-4}$, for robustly estimating efficient portfolio weights:

$$\mathbf{w}^{(j)} = \left(\hat{E}[\mathbf{R}_t^{(j)}\mathbf{R}_t^{(j)\prime}] + \gamma \mathbf{I}_{k+1}\right)^{-1} \hat{E}[\mathbf{R}_t^{(j)}], \tag{3}$$

where $\mathbf{I}_{k+1}$ is the identity matrix. Shrinkage parameters stabilize portfolio weight estimation and prevent over-leveraging, which can be adjusted to control the degree of regularization.

[10] We use a small shrinkage parameter, $\gamma_f = 10^{-5}$, for estimating MVE portfolio weights on factors.


$R_t^{(0)}$: SUE ≤ -0.2
Yes
No
$R_{1,t}^{(1)}$: DOLVOL ≤ -0.6
$R_{3,t}^{(2)}$
Yes
No
$R_{1,t}^{(2)}$
$R_{2,t}^{(2)}$
$R_t^{(0)}$: SUE ≤ -0.2
Yes
No
$R_{1,t}^{(2)}$
$R_{2,t}^{(1)}$: DOLVOL ≤ -0.6
Yes
No
$R_{2,t}^{(2)}$
$R_{3,t}^{(2)}$

(a) Splitting node $R_{1,t}^{(1)}$ at DOLVOL.

(b) Splitting node $R_{2,t}^{(1)}$ at DOLVOL.

Fig. 3. Illustration of the second split.
There are two potential candidates for the second split, each splitting one of the original leaf nodes. After the second split, three leaf basis portfolios are produced to update the latent factor.

Block structure in boosted P-Trees. The boosted P-Tree utilizes a two-step process to produce either the MVE portfolio or the SDF by incorporating multiple sets of leaf basis portfolios. Initially, the tree-specific MVE portfolio of all leaf basis portfolios within each P-Tree is computed to determine the current P-Tree factor. Subsequently, this factor is merged with all previous P-Tree factors or benchmarks to create an all-tree MVE portfolio for the multi-factor P-Tree model. Boosted P-Trees indirectly impose regularization on the all-tree MVE portfolio weights through the sequential tree block structure, even though the resulting Sharpe ratio may not surpass the one generated directly from all basis portfolios in all P-Trees. This avoids high-dimensional estimation issues and, in turn, leads to robust estimation and OOS performance.

2.4. Simulation

Simulation exercises in this section highlight the importance of using out-of-sample tests and of including most available observable features when applying P-Tree, as well as demonstrate the ability of P-Tree to capture true underlying characteristics generating the data and their non-linear interactions. For more details, please refer to Internet Appendix I.

We introduce three true underlying characteristics, their interactions, and nonlinear terms (detailed in Internet Appendix I). We calibrate the return-generating process on historical data (introduced in the next section) and include a large set of redundant characteristics correlated with the true characteristics generating the data, but do not drive returns. We then conduct the following four sets of exercises.

First, we compare the Sharpe ratios and alphas of P-Tree test assets with conventional characteristics-sorted portfolios, which do not incorporate characteristic interactions. The single P-Tree and boosted P-Trees consistently produce leaf basis portfolios with higher Sharpe ratios and larger unexplained pricing errors under various signal-to-noise-ratio levels, both in- and out-of-sample. Even when sorting is performed on the true characteristics directly, univariate- and bivariate-sorted portfolios still underperform P-Tree test assets generated on high-dimensional characteristics. This result is corroborated later in our empirical findings that P-Tree test assets achieve better mean–variance efficiency than sorted portfolios (Section 4.2). P-Tree indeed captures important interactions in the data-generating process.

Second, the interpretability of one P-Tree or a few P-Trees relies on their ability to select variables that matter. P-Tree helps distinguish sparse characteristics underlying the true data generation from a large number of redundant or useless ones. With repeated simulations, P-Tree selects the true characteristics in the first few splits with high probabilities. These improved selection rates are particularly evident in scenarios with a high signal-to-noise ratio. This discovery can only be discerned through simulations, as real-world data's sparse characteristics are still unknown. The consistent selection behavior relates to assessing characteristic importance in Section 5.1.

Third, we break down the difference between the in- and out-of-sample performance of P-Tree models into two components: *overfitting* and *limits to learning*, as discussed by Didisheim et al. (2024). Estimating the *true predictability* is challenging in real data due to the unknown market data-generating process. In our simulation, however, the true predictability is defined as the OOS Sharpe ratio achieved by P-Trees with oracle characteristics. We find positive values for overfitting and limits to learning, which increase with more boosted P-Trees, supporting the conclusion in Didisheim et al. (2024) that overfitting and limits to learning become more problematic with additional model parameters and a limited number of observations. Later in empirical analyses, we observe again the large gap between the in- and out-of-sample performance of P-Tree models, justifying our focus on OOS metrics.

Fourth, we evaluate the efficiency loss when a P-Tree misses some true characteristics among input features (but redundant or useless characteristics abound). This exercise evaluates the model's performance with incomplete or useless information, which may occur for various reasons. Under our calibrated return-generating process, the Sharpe ratios and CAPM alphas decline substantially, and P-Tree selects redundant or useless characteristics more frequently. Therefore, we should include all available predictors. In our later analyses of historical data, we indeed use all firm characteristics available in the dataset.

3. Splitting the cross section of U.S. equities

The P-Tree framework easily applies to public equities or corporate bonds. We focus on U.S. equities for an illustrative application. Notably, against the backdrop of emergent large models, P-Tree offers an interpretable alternative that does not require excessive computation resources.[11]

3.1. Data on U.S. public equities

Equity data and characteristics. The standard filters (e.g., same as in Fama–French factor construction) are applied to the universe of U.S. equities. This universe includes only stocks listed on NYSE, AMEX, or NASDAQ for more than one year and uses those observations for firms with a CRSP share code of 10 or 11. We exclude stocks with negative

[11] The baseline single P-Tree model trained on U.S. data runs about 20 min on a server with an Intel Xeon Gold 6230 CPU, for a training dataset with 61 characteristics and 2.2 million observations.

Table 1
Equity characteristics.
This table lists the description of 61 characteristics used in the empirical study.

No.	Char.	Description	Category
1	ABR	Abnormal returns around earnings announ.	Momentum
2	ACC	Operating accruals	Investment
3	ADM	Advertising expense-to-market	Intangibles
4	AGR	Asset growth	Investment
5	ALM	Quarterly asset liquidity	Intangibles
6	ATO	Asset turnover	Profitability
7	BASPREAD	Bid–ask spread (3 months)	Frictions
8	BETA	Beta (3 months)	Frictions
9	BM	Book-to-market equity	Value-versus-growth
10	BM_IA	Industry-adjusted book to market	Value-versus-growth
11	CASH	Cash holdings	Value-versus-growth
12	CASHDEBT	Cash to debt	Value-versus-growth
13	CFP	Cashflow-to-price	Value-versus-growth
14	CHCSHO	Change in shares outstanding	Investment
15	CHPM	Change in Profit margin	Profitability
16	CHTX	Change in tax expense	Momentum
17	CINVEST	Corporate investment	Investment
18	DEPR	Depreciation/PP&E	Momentum
19	DOLVOL	Dollar trading volume	Frictions
20	DY	Dividend yield	Value-versus-growth
21	EP	Earnings-to-price	Value-versus-growth
22	GMA	Gross profitability	Investment
23	GRLTNOA	Growth in long-term net operating assets	Investment
24	HERF	Industry sales concentration	Intangibles
25	HIRE	Employee growth rate	Intangibles
26	ILL	Illiquidity rolling (3 months)	Frictions
27	LEV	Leverage	Value-versus-growth
28	LGR	Growth in long-term debt	Investment
29	MAXRET	Maximum daily returns (3 months)	Frictions
30	ME	Market equity	Frictions
31	ME_IA	Industry-adjusted size	Frictions
32	MOM12M	Cumulative returns in the past (2-12) months	Momentum
33	MOM1M	Previous month return	Momentum
34	MOM36M	Cumulative returns in the past (13-35) months	Momentum
35	MOM60M	Cumulative returns in the past (13-60) months	Momentum
36	MOM6M	Cumulative returns in the past (2-6) months	Momentum
37	NI	Net equity issue	Investment
38	NINCR	Number of earnings increases	Momentum
39	NOA	Net operating assets	Investment
40	OP	Operating profitability	Profitability
41	PCTACC	Percent operating accruals	Investment
42	PM	Profit margin	Profitability
43	PS	Performance Score	Profitability
44	RD_SALE	R&D-to-sales	Intangibles
45	RDM	R&D-to-market	Intangibles
46	RE	Revisions in analysts' earnings forecasts	Intangibles
47	RNA	Return on net operating assets	Profitability
48	ROA	Return on assets	Profitability
49	ROE	Return on equity	Profitability
50	RSUP	Revenue surprise	Momentum
51	RVAR_CAPM	Idiosyncratic volatility - CAPM (3 months)	Frictions
52	RVAR_FF3	Res. var. - Fama–French 3 factors (3 months)	Frictions
53	SVAR	Return variance (3 months)	Frictions
54	SEAS1A	1-Year Seasonality	Intangibles
55	SGR	Sales growth	Value-versus-growth
56	SP	Sales-to-price	Value-versus-growth
57	STD_DOLVOL	Std of dollar trading volume (3 months)	Frictions
58	STD_TURN	Std. of Share turnover (3 months)	Frictions
59	SUE	Standardized unexpected quarterly earnings	Momentum
60	TURN	Shares turnover	Frictions
61	ZEROTRADE	Number of zero-trading days (3 months)	Frictions

book equity or lag market equity. We use 61 firm characteristics with monthly observations for each stock, covering six major categories: momentum, value, investment, profitability, frictions (or size), and intangibles. Characteristics are standardized cross-sectionally to the range $[-1, 1]$. Table 1 lists these input variables.[12]

The monthly data ranges from 1981 to 2020. The average and median monthly stock observations are 5265 and 4925 for the first 20 years, and 4110 and 3837 for the latter 20 years. We apply cross-sectional winsorization at 1% and 99% to mitigate the impact of outliers on individual stock returns. The entire 40-year sample is used for benchmark analysis. The sample is split into two parts – the first 20 years from 1981 to 2000 and the recent 20 years from 2001 to 2020 – for a subsample robustness check and as training and test samples.

Table 2
Macroeconomic variables.
This table lists the description of macro variables used in the empirical study.

No.	Variable name	Description
1	DFY	Default yield
2	DY	Dividend yield of S&P 500
3	EP	Earnings-to-price of S&P 500
4	ILL	Pastor-Stambaugh illiquidity
5	INFL	Inflation
6	LEV	Leverage of S&P 500
7	NI	Net equity issuance of S&P 500
8	SMVAR	Stock Market (S&P 500) Volatility
9	TBL	Three-month treasury bill rate
10	TMS	Term spread

Macroeconomic variables. In addition, we use 10 macro variables to capture potential regime switches. Table 2 summarizes the macro variables, which include market timing macro predictors, bond market predictors, and aggregate characteristics for S&P 500. We standardize these macro predictor data by the historical percentile numbers for the past 10 years.[13] This rolling-window data standardization is useful when comparing the predictor level to detect different macroeconomic regimes.

3.2. Visualizing a single P-Tree

Fig. 4 plots the P-Tree diagram. In each leaf node, S# represents the order of sequential splits, and N# is the node index. We provide the numbers in the terminal leaves for the portfolio size: the monthly median number of stock observations. The split rules are the selected splitting characteristics and cross-sectional quintile cutpoints $[-0.6, -0.2, 0.2, 0.6]$. Before the first split, the tree grows from the root node (N1), whose leaf basis portfolio represents the value-weighted market portfolio.

The data-driven P-Tree first splits along the standardized unexpected quarterly earnings (SUE, Rendleman et al., 1982) at -0.2 (40% percentile). After this split, 40% of the stocks go to the left leaf (labeled N2), and 60% go to the right (N3). Then, the second split is on the dollar trading volume (DOLVOL, Chordia et al., 2001) at -0.6 of the right leaf (N3), and the third split is also on DOLVOL at -0.6 of the left leaf (N2). Furthermore, subsequent splits include the industry-adjusted book-to-market ratio (BM_IA), return-on-equity (ROA), zero trade (ZEROTRADE), and market equity (ME). After nine splits, we stop the P-Tree growth and obtain 10 leaf basis portfolios.

Clustering patterns and asymmetric interactions. P-Tree clusters similar assets based on underlying characteristics, revealing sources of mean–variance diversification. Fig. 4 shows the (asymmetric) interactions of characteristics for splitting the cross section. By jointly defining the partition corresponding to the leaf node, P-Tree learns the interaction of characteristics appearing in the same path. For instance, ZEROTRADE of liquidity (Liu, 2006) is a valuable indicator for further splitting low-SUE low-DOLVOL stocks. However, for low-SUE non-low-DOLVOL stocks, ME of size (Banz, 1981) might be a better indicator for a further split under the MVE framework.

[12] For example, market equity values in Dec. 2020 are uniformly standardized into $[-1, 1]$. The firm with the lowest value is -1, and the highest is 1. All others are distributed uniformly in between. Missing values of characteristics are imputed as 0, which implies the firm is neutral in the security sorting.

[13] For example, inflation greater than 0.7 implies the current inflation level is higher than 70% of observations during the past decade.

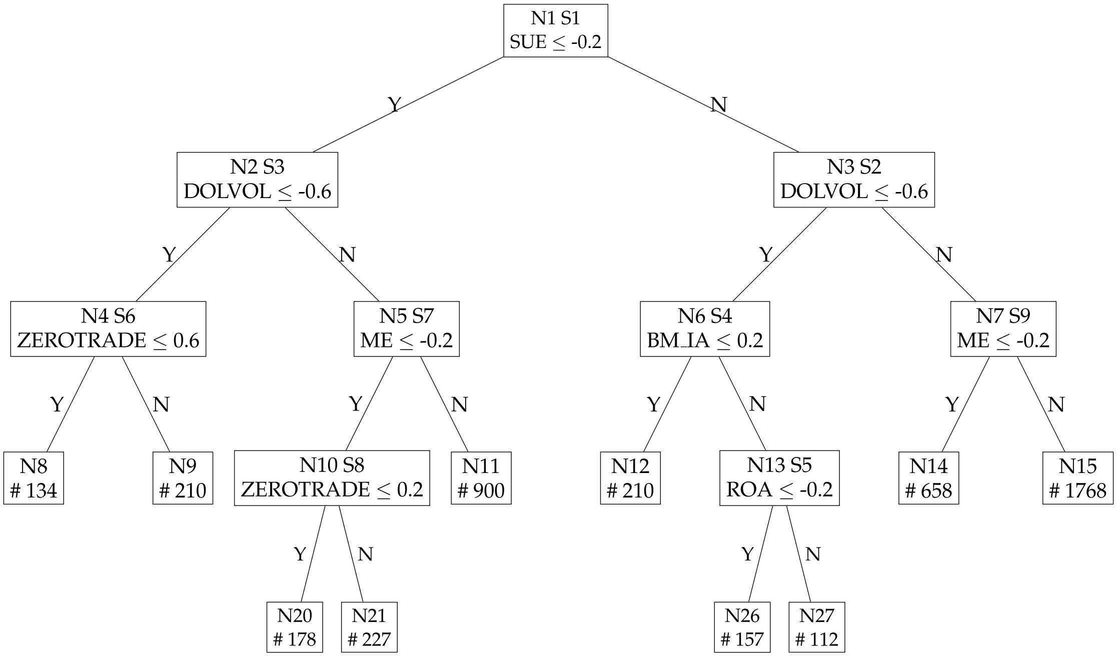


Fig. 4. Panel tree from 1981 to 2020.

We provide splitting characteristics, cutpoint values for each parent node, and their respective node and splitting indexes. For example, node N1 is split by SUE at -0.2 (40% percentile) as the first split S1, and the second split S2 is on node N3 by DOLVOL at -0.6 (20% percentile). The median monthly number of assets in the terminal leaf basis portfolios is also included. For example, node N8 has 134 stocks by monthly median. Table 1 describes equity characteristics. Fig. A.1 reports the P-Tree diagrams for subsamples.

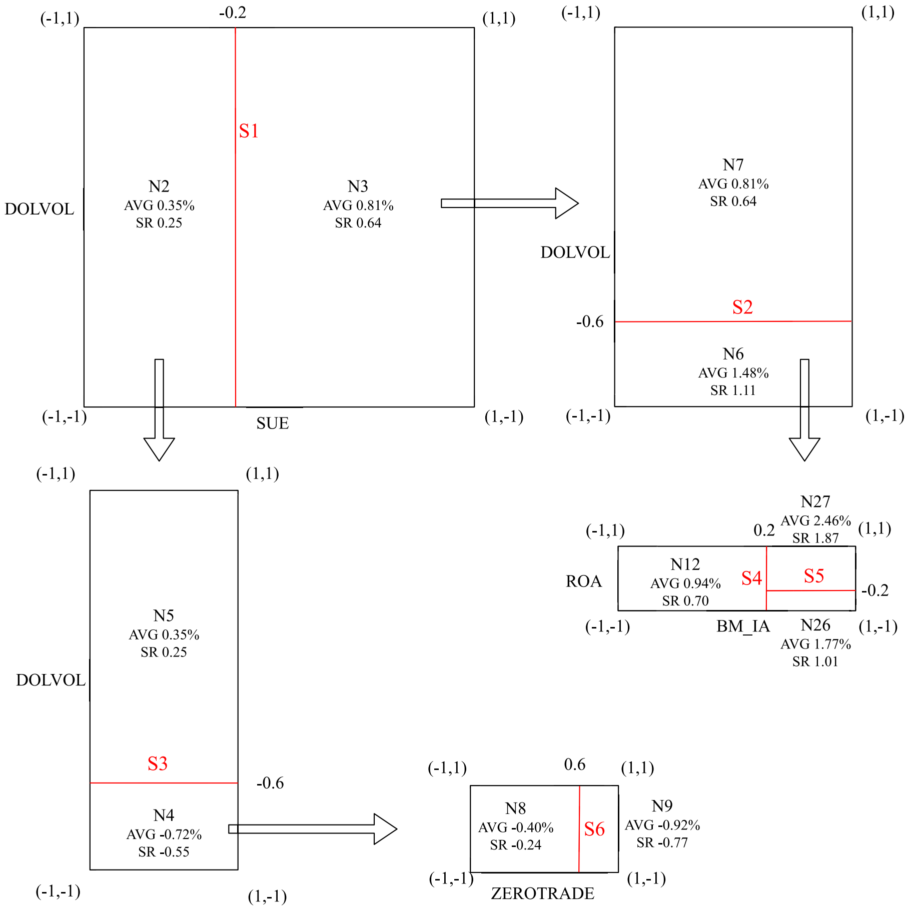


Fig. 5. Visualizing nonlinear interactions with partition plots.

This diagram illustrates the partitions for the first few splits of the tree structure in Fig. 4. The first split (S1) occurs at 40% of SUE on the entire stock universe, and the second split occurs at 80% of DOLVOL on the high SUE partition. The portfolio results for each partition are provided (monthly average excess returns (AVG) and annualized Sharpe ratios (SR)). The arrows indicate the next split is implemented on the partitioned area from the previous one.

The current literature has recognized interactions between two characteristics, but a more systematic investigation is needed. For instance, Lee and Swaminathan (2000) enhance momentum strategies by interacting with trading volume, and Da et al. (2014) find that the momentum effect is stronger for firms with continuous information than firms with discrete information. Our framework allows for exploring interactions among multiple characteristics, going beyond the typical bivariate relations. This is demonstrated in Fig. 4, where interaction paths involving at least four characteristics, such as SUE, DOLVOL, ME, and ZEROTRADE, can be identified.

The partition plot in Fig. 5 is an alternative way to visualize the clustering and asymmetric patterns. We report each leaf's monthly average excess returns and annualized Sharpe ratios. First, P-Tree splits on SUE at −0.2, which yields a low-SUE portfolio including 40% of the stocks and a high-SUE portfolio constituted by 60% of the stocks in the cross section. The spread of monthly expected returns between N2 and N3 is 0.46%, and the Sharpe ratios range from 0.25 to 0.64.

Second, we split on DOLVOL at −0.6 on N3 to harvest N6 and N7, where the low-DOLVOL portfolio N6 has *higher* expected returns and Sharpe ratio than N7. Third, we split on DOLVOL at −0.6 on N2 to harvest N4 and N5, where the low-DOLVOL portfolio N4 has *lower* expected returns and Sharpe ratio than N5. We find DOLVOL has different impacts on N2 and N3. On the high-SUE side, DOLVOL positively correlates to asset returns. However, on the low-SUE side, DOLVOL negatively correlates with asset returns. This is an example of asymmetric interaction between SUE and DOLVOL. A simple trading strategy that shorts N4 (Low-SUE -Low-DOLVOL portfolio) and longs N6 (High-SUE - Low-DOLVOL portfolio) makes over 2% monthly expected return. The return gaps among the leaves show the usefulness of splitting the cross section via the asymmetric interaction of characteristics.

3.3. P-Tree leaf basis portfolios

Asset clustering. P-Tree generates leaves, a.k.a., leaf basis portfolios. Unlike the scalar output in CART for pure prediction, a leaf of P-Tree represents a time series of portfolio returns from a time-invariant P-Tree structure. These leaf basis portfolios are nonlinear, interactive, and high-dimensional characteristics-managed portfolios. Table 3, panel A, summarizes leaf basis portfolios generated by the first P-Tree in Fig. 4: the index of nodes, median number of stocks in the leaf basis portfolio, average returns, CAPM α (%), β, time series regression R^2, and alphas with respect to FF5, Q5, RP-PCA, and IPCA five-factor models (Fama and French, 2015; Hou et al., 2021; Kelly et al., 2019; Lettau and Pelger, 2020).

We observe two large leaves: N11 containing 900 stocks and N15 containing over 1700 stocks by monthly median. Their CAPM R^2 are over 90%, and the β are close to one, meaning their return time series are highly correlated with the market returns. No further splitting results in higher Sharpe ratios on these two leaves, yet other leaves offer higher investment improvement under the MVE framework. For instance, N21 represents the low-SUE, high-DOLVOL, low-ME, and high-ZEROTRADE leaf, which has −1.14% average return and −1.83% CAPM α, and N27 is the high-SUE, low-DOLVOL, high-BM_IA, and high-ROA leaf, which has 2.46% average returns and 1.97% CAPM α. By employing a simple monthly rebalanced strategy of buying N27 and selling short N21, one can expect an average return of 3.6%.

These leaf basis portfolios are generated via interactive and nonlinear splits on 61 characteristics. We expect them to be very hard to price using the prevalent linear factor models. Nine leaves have significant CAPM alphas among the 10 leaf basis portfolios. At a confidence level of 10%, the number of statistically significant non-zero alphas for FF5, Q5, RP-PCA 5, and IPCA 5 factor models are 9, 9, 9, and 7, respectively.

P-Tree is a goal-oriented clustering algorithm that maximizes the Sharpe ratio in the baseline specification. Therefore, we anticipate these test assets having economically large and statistically significant alphas against those "inadequate" benchmark factor models. With such distinct basis portfolios, we expect them to achieve higher mean–variance efficiency, spanning an efficient frontier that benchmark factor models fail to span. Panels B and C of Table 3 provide subsample analysis for the first and the latter 20 years. Although the tree structure and leaf identity change because of refitting, we still find similar patterns. For example, pricing these portfolios based on leaf factors remains difficult using benchmark factor models.

Diversified test assets. Fig. 6, Panel A, depicts the performance statistics of P-Tree test assets. Subfigure A1 shows the mean-standard-deviation scatter plot for leaf basis portfolios of the first P-Tree on the 40-year sample from 1981 to 2020 in black circles. The expected returns are in the range of −2% to 3%, and the standard deviations are in the range of 4% to 9%. We observe large variations in these portfolios' expected returns and risks, which means they are diversified under the MVE framework. Significant variations exist among the portfolios in Panels A2 and A3 for subsample analysis, with the earlier sample showing more diversification and the latter sample being more concentrated. By contrast, the 5 × 5 ME-BM portfolios in light-red triangles cluster around the same level of average returns, which are much less diversified than those of P-Tree test assets. This finding indicates that P-Tree test assets are more diversified under the MVE framework than the ME-BM portfolios.

Panel B of Fig. 6 shows the CAPM model-implied alpha-beta plots. In subfigure B1's 40-year sample, we find CAPM βs of P-Tree test assets are scattered around one in the range of 0.6 to 1.6. CAPM alphas are around zero, ranging from −2% to 2%. These basis portfolios have a lot of variations in CAPM alphas and βs. The alpha and beta ranges were larger in the first 20 years in B1, whereas the variation was smaller in the second 20 years in B2. The 5 × 5 ME-BM portfolios are more easily explained for the CAPM model than P-Tree test assets because they align closely with the vertical line of zero alpha with more negligible diversification in all figures. In summary, this is highly positive evidence for P-Tree test assets' better-diversified patterns than conventional ones, even when priced by CAPM.

4. Boosted P-Trees for multiple factors and test assets

4.1. Growing the efficient frontier

Our investigation focuses on the efficient frontiers spanned by test assets generated using (boosted) P-Tree. Many P-Tree test assets have significant alphas over multiple well-known benchmark factor models, indicating those factor models cannot effectively span the efficient frontier. The boosting technique enhances our ability to generate more effective test assets that expand the efficient frontier better, that is, "growing the efficient frontiers on P-Trees".

Unspanned efficient frontier. Fig. 7 shows the MVE frontiers of P-Trees and boosted P-Trees in gradient color from blue to red for the sample period from 1981 to 2020. Benchmark factor models are printed in black, and benchmark test assets in purple. Notably, the P-Tree1 frontier (generated by the first P-Tree test assets) is already more efficient than any frontier spanned by the benchmark factor models and test assets. Among the benchmark test assets, 285 bivariate-sorted portfolios span a frontier very close to our P-Tree1 frontier. However, P-Tree1 contains only 10 portfolios, much smaller than 285. Among the factor models, the IPCA five-factor model is a strong candidate but still less efficient than the P-Tree1 frontier. Overall, these 10 leaf basis portfolios consist of promising test assets to span the efficient frontier, which is more efficient than the benchmarks under the MVE framework. In other words, the listed benchmark factor models and test assets are insufficient to span the real efficient frontier.

Table 3
Evaluation for leaf basis portfolios.
This table reports the properties of 10 leaf basis portfolios generated by the first P-Tree: the median number of stocks (leaf size), average returns, CAPM α (%), β, R^2, and the alphas (%) with respect to FF5, Q5, RP-PCA, and IPCA five-factor models. Panel A corresponds to Fig. 4, and Panels B and C correspond respectively to (a) and (b) in Fig. A.1.

ID	# Median	AVG	STD	α_{CAPM}	β_{CAPM}	R^2_{CAPM}	α_{FF5}	α_{Q5}	α_{RP5}	α_{IP5}
Panel A: 40 Years (1981–2020)										
N8	134	−0.40	5.76	−0.97***	0.84	0.42	−0.91***	−0.66***	−1.51***	−0.58*
N9	210	−0.92***	4.14	−1.35***	0.63	0.47	−1.46***	−1.25***	−1.84***	−0.36*
N20	178	−0.34	8.89	−1.34***	1.45	0.53	−1.10	−0.49*	−1.55***	−1.78***
N21	227	−1.14***	5.90	−1.83***	1.00	0.58	−1.83***	−1.56***	−2.34***	−1.19***
N11	900	0.36*	4.81	−0.35***	1.04	0.93	−0.30***	−0.15**	−0.29***	0.16
N12	210	0.94***	4.63	0.42**	0.76	0.54	0.29**	0.48***	−0.22	1.17***
N26	157	1.77***	6.04	1.16***	0.88	0.43	1.08***	1.37***	0.40**	1.40***
N27	112	2.46***	4.55	1.97***	0.71	0.49	1.80***	2.00***	1.18***	2.18***
N14	658	1.03***	6.88	0.21	1.18	0.59	0.32*	0.64***	−0.33**	0.06
N15	1768	0.81***	4.41	0.14***	0.98	0.98	0.10***	0.03	−0.26***	0.29
Panel B: 20 Years (1981–2000)										
N16	97	0.25	5.99	−0.34	0.81	0.36	−0.27	−0.08	−0.27	−0.02
N17	256	−0.74***	4.40	−1.16***	0.57	0.33	−1.36***	−1.17***	−1.56***	−1.47***
N18	95	1.72***	5.72	1.17***	0.75	0.34	1.07***	1.27***	0.84**	0.03
N19	48	2.91***	4.97	2.44***	0.64	0.33	2.13***	2.50***	2.36***	1.87***
N5	3443	0.76***	4.39	0.03*	0.99	1.00	−0.02*	−0.04***	−0.49***	1.34***
N24	160	−3.53***	7.75	−4.36***	1.13	0.42	−3.90***	−3.29***	−3.58***	0.79
N25	250	−2.82***	6.85	−3.58***	1.04	0.45	−3.20***	−2.54***	−1.89***	1.55*
N13	286	−1.75***	9.94	−2.86***	1.52	0.46	−1.58***	−1.19***	−1.29***	2.22**
N14	217	−1.21*	9.60	−2.21***	1.37	0.40	−1.16***	−0.63	−0.09	4.52**
N15	116	1.23*	10.07	0.16	1.46	0.41	1.12***	1.52***	1.39**	3.58
Panel C: 20 Years (2001–2020)										
N8	153	−0.20	4.44	−0.63**	0.67	0.45	−0.63**	−0.46**	−0.93***	−0.10
N18	58	−0.31	6.82	−0.96***	1.01	0.44	−0.93***	−0.53*	−1.36***	−0.88**
N19	112	−0.90***	4.72	−1.39***	0.76	0.53	−1.39***	−1.31***	−1.89***	−0.93***
N5	1206	0.35	5.23	−0.38***	1.13	0.95	−0.27***	−0.25***	−0.21***	0.37*
N24	171	0.94**	6.47	0.26	1.05	0.53	0.31	0.57**	−0.13	0.59**
N25	104	2.44***	6.80	1.79***	1.02	0.45	1.88***	2.17***	1.16***	1.43***
N52	47	1.68***	5.28	1.17***	0.79	0.46	1.17***	1.45***	0.52**	1.56***
N53	76	3.53***	5.27	2.97***	0.86	0.54	2.88***	3.10***	2.24***	3.19***
N27	76	1.79***	4.05	1.36***	0.66	0.53	1.31***	1.37***	0.92***	1.6***
N7	1896	0.76***	4.27	0.16***	0.94	0.98	0.10**	0.09**	−0.05	0.63***

*, **, and *** represent significance levels of 10%, 5%, and 1%, respectively.

Boosting the empirical frontier. Beyond the first P-Tree, boosted P-Trees improve portfolio efficiency from one factor to 20 factors. In Fig. 7, the spanned frontiers move toward the top left corner of the mean–variance diagram. Our approach consistently advances the efficient frontier throughout the 20 P-Trees. Combining 200 assets in one MVE portfolio is challenging because of the curse of dimensionality. Unlike Ait-Sahalia and Xiu (2017), who use exogenous industry classifications to construct block-diagonal patterns, our boosted P-Trees entertain an endogenous block-diagonal structure in the return covariance. We combine the 10 leaf basis portfolios of each P-Tree into a single P-Tree factor and then estimate the tangency portfolio of the boosted P-Tree factors. As such, we can generate multiple factors and combine multiple sets of test assets in one frontier under the MVE framework.

Moving from a single P-Tree to 20 P-Trees results in more efficient frontiers, allowing investors to achieve higher expected returns while bearing the same level of risk. Although a significant difference exists between the frontiers of P-Tree1 and P-Tree10, the frontier lines increase slowly after P-Tree10, indicating a more negligible improvement in mean–variance efficiency with each subsequent boosting.

We employ asset pricing tests to demonstrate the incremental efficiency of boosted P-Trees. First, we show the Sharpe ratios of each boosted P-Tree factor and the multi-factor model. Second, we evaluate each boosted factor with CAPM and FF5 by checking for a significant alpha. As we know from Table 3, P-Tree test assets are hard to price, and we expect the P-Tree factors to pass the CAPM test and FF5 test. Third, we regress each P-Tree factor on all the previous boosted P-Tree factors generated for the expanding factor tests. The efficiency of the frontier increases if the previous P-Tree factors do not span the additional P-Tree factor.

The test results are presented in Table 4. The Sharpe ratios of each boosted factor are above 1, even for the 20-th factor. Meanwhile, the cumulative multi-factor Sharpe ratio increases monotonically from 6 to 15, which means that the test assets of the boosted P-Tree add incremental pricing information to the existing test assets. Furthermore, all CAPM and FF5 alphas are positive and highly significant, indicating that all the boosted P-Tree factors cannot be explained by CAPM or FF5. In the expanding factor test, we find significant alphas, large t-statistics, and high time series R^2. After applying the nested asset pricing test (Barillas and Shanken, 2017), we do not find any evidence that supports a sufficient factor model to explain the assets in the boosted P-Tree test.[14] In summary, Table 4 shows that boosted P-Trees generate unexplained alphas, and their leaf portfolios thus better span the efficient frontier. By contrast, conventional univariate- or bivariate-sorted test assets used for testing asset pricing models do not fully cover the efficient frontier.

4.2. Generating test assets via boosted P-Trees

First, we show the pricing performance of leaf basis portfolios, implying they serve as diversified test assets and are challenging against

[14] The first row contains "-" signs, which indicate that the expanding factor and BS tests are not applicable for the single factor case.

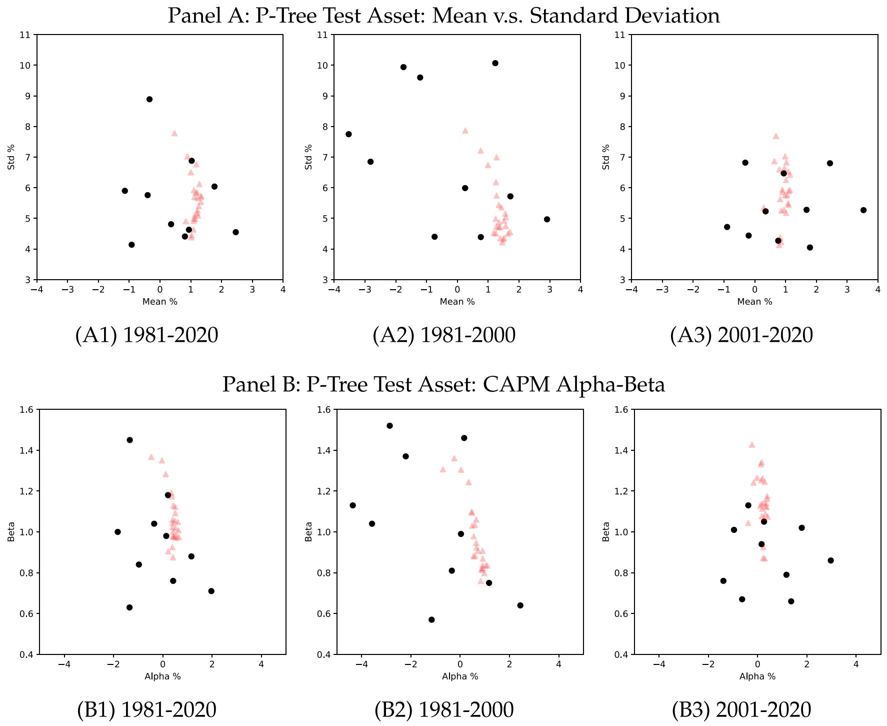


Fig. 6. Diversified P-Tree test assets.
This figure reports the performance of P-Tree test assets (presented in Table 3) and the 5 × 5 ME-BM portfolios. The 10 black circles represent P-Tree test assets, and the 25 light-red triangles represent ME-BM portfolios. Panel A shows the scatter plots of the mean and standard deviation for portfolio returns in percentage. Panel B shows the scatter plots of CAPM alpha and beta, with alphas in percentage.

the FF5 model. Then, we utilize boosted P-Trees that generate multiple sets of basis portfolios. Empirically, we grow up to 20 P-Trees, yielding 200 leaf basis portfolios.

P-Tree test assets. Table 5, Panel A, lists the number of portfolios, the GRS test statistic and its p-value, the p-value of PY test,[15] the average absolute α, the root mean squared α, the average time series regression R^2, and the percentage of significant alphas with respect to 10%, 5%, and 1% confidence levels. In the rows, P-Tree1 represents a set of test assets comprising 10 leaf basis portfolios in the first P-Tree. P-Tree1-5 combines all the leaf basis portfolios generated by the first five P-Trees, and P-Tree6-10 contains the 6-th to the 10-th P-Trees. Similarly, we define P-Tree11-15, P-Tree16-20, and P-Tree1-20.

We find the GRS test and PY test always reject the null hypothesis that the alphas of test assets are jointly zero for all specifications of P-Tree test assets, which means the expected returns of P-Tree test assets can hardly be explained by the FF5 model. Furthermore, the last three columns show that many assets have statistically significant alphas. For comparison, we include four sets of test assets that are commonly used in literature: 150 10 × 1 univariate-sorted portfolios, 285 bivariate-sorted portfolios, 5 × 5 ME-BM portfolios, and 49 industry portfolios, all of which can be downloaded from Ken French's website. P-Tree test assets have larger GRS test statistics and alphas than these benchmark test assets.

Furthermore, we observe the decline of alphas from the first P-Tree to the subsequently boosted P-Trees as anticipated, indicating diminishing marginal convergence to the limit of the efficient frontier. The test assets of the first P-Tree have larger GRS statistics, higher average alphas, and higher proportions of significant alphas than the test assets generated by the follow-up P-Trees. In addition, we see an evident decline in GRS statistics and alphas from P-Tree1-5 to P-Tree6-10, P-Tree11-15, and P-Tree16-20. P-Tree generates the most informative test assets in the first P-Tree, whereas the incremental information in the boosted P-Trees is complementary and declining. Although the alphas decline along boosting, the test assets generated by P-Tree16-20 cannot be explained by FF5, with a 4.21 GRS test statistic, 0.31% mean absolute alpha, and 52% test assets having significant alphas at the 10% level. The test assets of P-Tree16-20 are still harder to price than univariate- and bivariate-sorted portfolios.

Overall, these 200 test assets created by 20 P-Trees on high-dimensional characteristics challenge the FF5 model. They are more difficult to price than benchmark test assets, setting a high standard for testing factor models, which respond to the concerns in empirical studies (Lewellen et al., 2010; Ang et al., 2020).

For robustness, Table 5, Panels B and C, report the first and the latter 20-year subsamples. In the recent 20 years, FF5 has performed well in pricing the benchmark test asset; that is, the GRS and PY tests cannot be rejected at the 10% level, and less significant alphas testing on 49 industry portfolios and univariate-sorted portfolios. However, the P-Tree test assets always reject the null hypothesis for GRS and PY tests and are consistently challenging to price in both subsamples.[16]

Advantages of P-Tree clustering. P-Tree has two advantages that make these test assets efficient. First, P-Tree is a goal-oriented clustering algorithm tailored for spanning the efficient frontier iteratively. Each leaf clusters individual asset returns based on similar characteristic values, aggregating as a time series of portfolio returns. The goal is to maximize the collective Sharpe ratio of the MVE portfolio spanned by the leaf basis portfolios. The economic objective of P-Tree is vastly different from the CART and other off-the-shelf ML models, which focus on statistical return prediction without economic utilities and cannot

[15] Pesaran and Yamagata (2023) adapts to cases where the number of assets being tested is larger than the number of periods, which Gibbons et al. (1989) cannot address.

[16] The GRS test is not applicable in the presence of more test assets than observations, as is the case for Bi-Sort in Panels B and C, marked with "-".

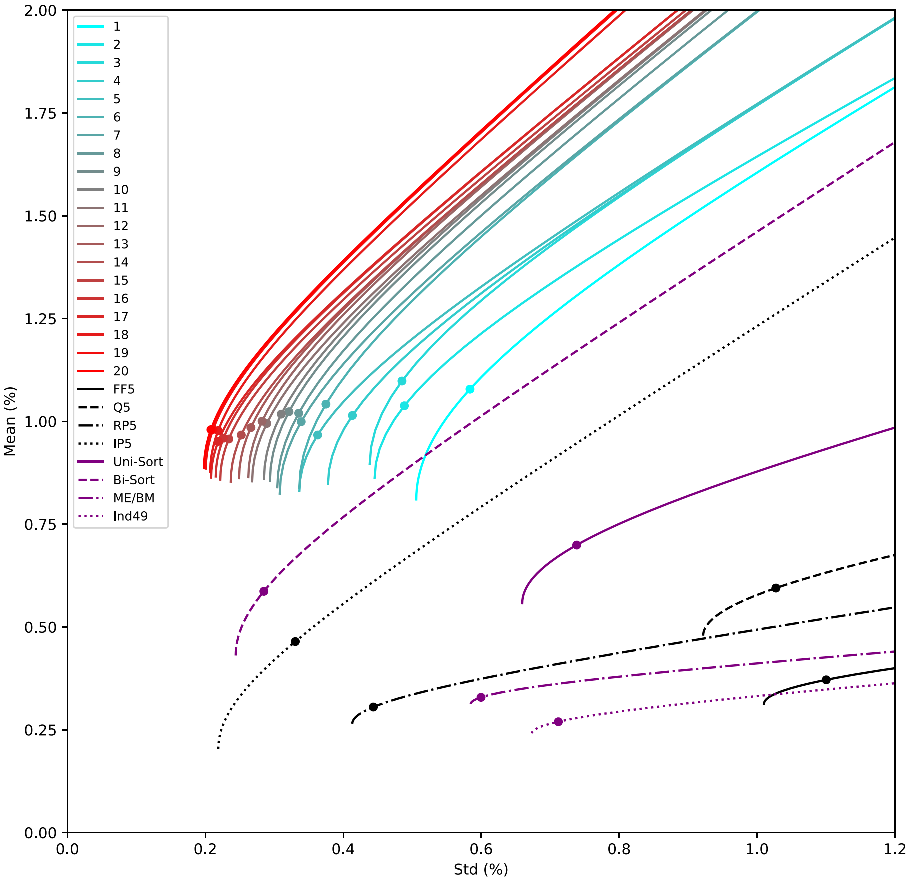


Fig. 7. Characterizing the efficient frontier with P-Trees.
This figure shows the MVE frontiers of the first P-Tree, the sequence of boosted P-Tree factor models, benchmark factor models, and benchmark test assets. The dots on the frontiers represent the tangency portfolios. The mean and standard deviation are on a monthly scale. The sample period is from 1981 to 2020.

ever tackle the task of growing an efficient frontier.

Second, P-Tree exploits the complex asymmetric interactions and nonlinearities on high-dimensional characteristics. The commonly used test assets are characteristics-sorted portfolios, such as univariate- and bivariate-sorted portfolios, which only consider decile or quintile sorting, along with up to three characteristics chosen ad hoc or based on researchers' domain expertise. In contrast, P-Tree is more flexible, allowing for general sequential and simultaneous sorting with more characteristics and asymmetric interactions in a unified framework. P-Trees enable the inclusion of large leaves, like N15, as shown in Fig. 4, where further splitting does not result in a higher Sharpe ratio, as well as small leaves, such as N8. The asymmetric splitting allows P-Trees to focus on the finer clusters of informative assets while paying less attention to non-informative assets under the MVE framework.

4.3. P-Tree factors and asset pricing performance

Using the findings from the previous analysis, we have established that P-Tree test assets are distinct from conventional test assets. Moving forward, we proceed with cross-examining P-Tree factors and various benchmark factor models on different sets of test assets, including P-Tree leaf basis portfolios. We adopt the cross-sectional R^2 to evaluate the asset pricing performance:

$$\text{Cross-sectional } R^2 = 1 - \frac{\sum_{i=1}^{N}\left(\bar{R}_i - \widehat{\bar{R}}_i\right)^2}{\sum_{i=1}^{N}\left(\bar{R}_i\right)^2}, \tag{6}$$

where $\bar{R}_i$ denotes the average return and $\widehat{\bar{R}}_i = \widehat{\beta}_i{}'\widetilde{\lambda}$ is the factor model implied average return of asset i. The risk premium estimation adopts the cross-sectional regression estimates of factors, and factor loadings are estimated from time series regressions. The cross-sectional R^2 represents the faction of assets' average returns explained by the model-implied expected returns. Various test assets, including P-Tree test assets, conventional sorted portfolios, and industry portfolios, are used to test the factor models. In Table 6, we display the factor models in the columns to explain the expected return of test assets in the rows. Panel A provides the asset pricing performances for the 40-year sample, and Panels B and C report for the 20-year subsamples.[17]

First, the P-Tree1 factor (from the first P-Tree) explains 98.3% of the cross-sectional return variation of the P-Tree1 leaf basis portfolios (10 assets), whereas the benchmark factor models provide much lower performance. This finding is not surprising, since the first P-Tree factor is constructed jointly with these test assets. However, the FF5, Q5, and RP-PCA five-factor models have much smaller R^2s than the P-Tree1 factor, indicating large pricing errors in explaining the expected returns of the highly diversified P-Tree1 test assets. One exception is that the IPCA five-factor model performs well in pricing the P-Tree1 test assets, with a 73.5% R^2.

Second, the P-Tree five-factor model (P-Tree1-5, read as a model containing from the first to the fifth factors), which includes five P-Tree factors from the first five P-Trees, demonstrates tremendous asset pricing performance. It outperforms the benchmark five-factor models in pricing the first 10 P-Trees' (P-Tree1-10) test assets and 5 × 5 ME-BM portfolios. For other specifications of test assets, P-Tree1-5 shows comparable levels of cross-sectional R^2 among all the five-factor

[17] The first row contains two "–,", which indicate the number of assets $N = 10$ is equal to or smaller than the number of factors (10 or 20). Thus, the cross-sectional regression is not applicable.

Table 4
Testing the boosted P-Tree growth.
This table shows the performance of each sequentially generated boosted P-Tree factor, including the Sharpe ratio, CAPM test, FF5 test, expanding factors test, and the (Barillas and Shanken, 2017) test. Panels A, B, and C report for different sample periods. The "Single" column displays the Sharpe ratio of the single factor, and the "Cumu." (cumulative) column shows the Sharpe ratio of the MVE spanned by multiple factors from the first to the current P-Tree factors. The columns report the α (%) and t-statistic for CAPM and FF5 spanning regression tests. For the expanding factor tests, we regress each P-Tree factor on all previously boosted P-Tree factors and report the α (%), t-statistic, and R^2.

	Sharpe ratio		CAPM test		FF5 test		Expanding factors test			BS test
	Single	Cumu.	α (%)	t-stat	α (%)	t-stat	α (%)	t-stat	R^2	p-value
Panel A: 40 Years (1981–2020)										
1	6.37	6.37	1.39	35.36	1.37	35.81	–	–	–	
2	3.20	7.35	0.52	17.65	0.48	15.27	0.62	9.55	0.01	0.00
3	1.18	7.80	0.34	5.25	0.18	3.28	−0.86	−5.43	0.26	0.00
4	2.06	8.46	0.44	11.22	0.38	9.56	0.69	7.63	0.16	0.00
5	1.99	9.18	0.48	10.25	0.41	9.25	0.84	5.61	0.21	0.00
6	1.01	9.57	0.18	4.24	0.08	2.94	−0.50	−5.47	0.45	0.00
7	1.42	10.11	0.28	7.62	0.22	6.67	0.63	7.03	0.36	0.00
8	1.32	10.40	0.28	7.14	0.20	4.95	−0.50	−5.33	0.41	0.00
9	1.83	10.88	0.53	10.11	0.43	9.38	0.85	6.88	0.34	0.00
10	1.48	11.20	0.44	7.35	0.30	7.30	−0.68	−5.49	0.46	0.00
11	1.78	11.72	0.38	10.33	0.31	8.62	0.72	6.70	0.30	0.00
12	1.02	12.06	0.20	4.68	0.10	2.99	−0.55	−5.76	0.55	0.00
13	1.37	12.57	0.29	8.38	0.22	6.48	0.76	6.01	0.33	0.00
14	1.37	13.01	0.48	5.93	0.32	5.86	−0.91	−5.68	0.60	0.00
15	1.37	13.81	0.31	6.58	0.21	5.66	0.97	7.11	0.48	0.00
16	1.24	14.28	0.28	6.23	0.17	4.11	−0.74	−6.24	0.54	0.00
17	1.54	14.60	0.46	8.16	0.40	7.52	−0.89	−4.80	0.40	0.00
18	1.64	14.92	0.32	8.43	0.27	7.78	−0.65	−5.43	0.34	0.00
19	1.48	15.43	0.43	8.63	0.36	7.47	1.18	5.86	0.34	0.00
20	1.35	15.63	0.33	7.38	0.23	6.19	−0.59	−4.19	0.44	0.00
Panel B: 20 Years (1981–2000)										
1	7.12	7.12	1.86	29.62	1.78	30.22	–	–	–	
2	1.09	7.75	0.25	3.66	0.11	1.91	−0.83	−4.80	0.28	0.00
3	2.26	9.45	0.39	9.72	0.34	8.48	0.92	8.19	0.17	0.00
4	6.48	11.35	1.80	26.96	1.76	27.06	1.75	10.16	0.03	0.00
5	1.59	12.71	0.33	6.50	0.28	5.77	1.11	6.85	0.36	0.00
6	1.65	13.67	0.37	5.87	0.26	4.57	−1.03	−5.34	0.40	0.00
7	1.72	14.94	0.33	6.99	0.32	6.17	1.25	5.67	0.20	0.00
8	1.52	15.78	0.39	6.10	0.31	4.93	−1.32	−5.12	0.35	0.00
9	3.02	17.90	1.03	14.00	0.93	13.08	2.69	8.42	0.26	0.00
10	1.93	19.18	0.83	8.12	0.67	7.14	−2.41	−6.27	0.52	0.00
11	1.34	20.72	0.38	4.93	0.19	2.76	−2.18	−7.33	0.45	0.00
12	2.22	22.89	0.36	9.38	0.31	8.03	1.46	7.36	0.39	0.00
13	1.51	24.14	0.35	4.88	0.23	3.69	1.82	6.06	0.43	0.00
14	2.11	25.71	0.58	7.80	0.48	7.57	−2.43	−5.93	0.40	0.00
15	2.39	28.37	0.87	10.14	0.76	8.70	3.69	7.64	0.52	0.00
16	2.53	30.37	0.53	9.58	0.44	7.91	2.40	7.12	0.31	0.00
17	1.50	32.42	0.30	5.66	0.31	5.08	2.65	7.18	0.35	0.00
18	1.63	33.89	0.37	5.95	0.28	4.99	−2.79	−5.64	0.37	0.00
19	1.84	35.78	0.47	7.04	0.37	5.93	3.54	7.35	0.30	0.00
20	2.70	37.94	0.89	9.85	0.79	9.06	−4.67	−6.91	0.44	0.00
Panel C: 20 Years (2001–2020)										
1	5.82	5.82	1.51	24.22	1.47	24.26	–	–	–	
2	2.14	6.64	0.58	8.74	0.52	8.26	0.91	6.44	0.03	0.00
3	0.91	7.53	0.18	3.28	0.09	2.01	−0.86	−5.82	0.36	0.00
4	1.35	8.60	0.29	6.18	0.31	6.04	1.00	8.65	0.17	0.00
5	1.08	9.30	0.31	4.14	0.26	3.88	−1.02	−6.38	0.42	0.00
6	2.63	10.36	0.60	12.15	0.56	11.11	1.04	7.61	0.12	0.00
7	1.39	11.30	0.29	5.53	0.26	5.16	1.00	7.23	0.28	0.00
8	1.60	11.93	0.40	5.76	0.30	5.47	−0.85	−5.62	0.44	0.00
9	1.04	12.73	0.21	4.53	0.18	3.83	−1.01	−7.44	0.36	0.00
10	1.64	14.32	0.30	5.96	0.25	5.31	1.11	7.80	0.44	0.00
11	1.69	15.73	0.36	6.73	0.30	6.20	1.43	6.79	0.33	0.00
12	1.58	16.82	0.60	6.08	0.51	5.36	2.48	5.72	0.24	0.00
13	1.48	18.06	0.41	5.29	0.29	4.47	−1.77	−5.52	0.47	0.00
14	1.39	19.46	0.32	5.22	0.26	4.62	1.68	6.05	0.40	0.00
15	1.48	20.60	0.35	5.51	0.21	4.34	−1.38	−6.25	0.58	0.00
16	1.40	21.91	0.34	5.84	0.27	4.82	1.84	6.29	0.46	0.00
17	1.27	23.41	0.39	4.82	0.25	3.46	−2.55	−6.99	0.51	0.00
18	1.44	24.36	0.34	5.16	0.28	4.72	−1.61	−4.97	0.48	0.00
19	1.58	25.45	0.33	6.59	0.28	6.08	−1.74	−5.20	0.42	0.00
20	1.88	26.52	0.58	7.12	0.50	6.53	−2.76	−4.96	0.39	0.00

Table 5
Comparing test assets.
The table displays performance statistics of P-Tree test assets and others for comparison, including P-Tree1, which consists of 10 portfolios in the first P-Tree. P-Tree1-5, P-Tree6-10, P-Tree11-15, P-Tree16-20, and P-Tree1-20 are sets of test assets that combine leaf basis portfolios generated by the first 5, 6-10, 11-15, 16-20, and 1-20 P-Trees, respectively. "Uni-Sort" has 150 univariate-sorted portfolios, "Bi-Sort" has 285 bivariate-sorted portfolios, ME/BE has 25 portfolios, and "Ind49" includes 49 industry portfolios. The reported statistics include the number of test assets, GRS test statistics and p-values (Gibbons et al., 1989), *p*-values of PY test (Pesaran and Yamagata, 2023), average absolute α (%), root mean squared α (%), average R^2 (%) of regressing the portfolios on Fama–French five factors, and the proportion of test assets with unexplained significant alphas under 10%, 5%, and 1% significance levels. The three panels report on the re-trained models for different sample periods.

	N	GRS	*p*-GRS	*p*-PY	$\overline{\lvert\alpha\rvert}$	$\sqrt{\overline{\alpha^2}}$	$\overline{R^2}$	$\%\alpha_{10\%}$	$\%\alpha_{5\%}$	$\%\alpha_{1\%}$
Panel A: 40 Years (1981–2020)										
P-Tree1	10	141.27	0.00	0.00	0.92	1.11	75	100	90	80
P-Tree1–5	50	60.32	0.00	0.00	0.44	0.61	80	70	62	44
P-Tree6–10	50	4.60	0.00	0.00	0.29	0.37	79	56	50	34
P-Tree11–15	50	4.74	0.00	0.00	0.20	0.26	80	38	36	24
P-Tree16–20	50	4.21	0.00	0.00	0.31	0.42	77	52	44	30
P-Tree1–20	200	41.31	0.00	0.00	0.31	0.43	79	54	48	33
Uni-Sort	150	1.62	0.00	0.00	0.10	0.14	88	25	18	7
Bi-Sort	285	2.50	0.00	0.00	0.12	0.17	89	30	23	15
ME-BM	25	5.01	0.00	0.00	0.12	0.16	92	36	28	20
Ind49	49	1.99	0.00	0.00	0.28	0.35	60	39	31	18
Panel B: 20 Years (1981–2000)										
P-Tree1	10	84.36	0.00	0.00	1.58	1.95	70	90	80	80
P-Tree1–5	50	50.84	0.00	0.00	0.79	1.26	76	62	60	52
P-Tree6–10	50	7.27	0.00	0.00	0.58	0.87	75	56	44	38
P-Tree11–15	50	6.39	0.00	0.00	0.55	0.82	76	66	60	42
P-Tree16–20	50	8.42	0.00	0.00	0.52	0.74	76	62	54	50
P-Tree1–20	200	112.90	0.00	0.00	0.61	0.95	76	62	55	46
Uni-Sort	150	1.94	0.00	0.00	0.17	0.23	88	35	27	19
Bi-Sort	285	–	–	0.00	0.21	0.30	89	42	33	21
ME-BM	25	4.75	0.00	0.00	0.21	0.25	91	56	48	32
Ind49	49	2.44	0.00	0.00	0.52	0.61	61	59	49	27
Panel C: 20 Years (2001–2020)										
P-Tree1	10	56.76	0.00	0.00	1.09	1.35	68	90	90	90
P-Tree1–5	50	30.35	0.00	0.00	0.43	0.68	76	52	38	24
P-Tree6–10	50	5.17	0.00	0.00	0.29	0.37	75	34	28	14
P-Tree11–15	50	2.20	0.00	0.00	0.27	0.35	75	30	22	10
P-Tree16–20	50	2.52	0.00	0.00	0.31	0.40	76	42	28	10
P-Tree1–20	200	83.91	0.00	0.00	0.33	0.47	76	40	29	14
Uni-Sort	150	1.46	0.03	0.94	0.09	0.12	89	12	7	0
Bi-Sort	285	–	–	0.01	0.11	0.15	91	21	15	6
ME-BM	25	2.58	0.00	0.10	0.11	0.14	93	24	8	8
Ind49	49	1.29	0.11	0.36	0.25	0.32	62	18	8	2

models for a fair comparison. The superior pricing performance remains consistent across Panels A to C for the entire sample and subsample analysis.

Third, these endogenously generated P-Tree test assets better expand the MVE frontier and raise the bar for testing asset pricing models. The commonly used test assets set a low bar for testing factor models, so benchmark factor models seem sufficient to price them. As we see from Panel A of Table 6, the listed benchmark five-factor models have over 90% cross-sectional R^2s on the benchmark test assets, and these numbers are consistently large in Panels B and C for subsample analyses. However, their R^2s decline dramatically when tested against P-Tree test assets. P-Tree test assets use nonlinear and asymmetric interactions among a large set of characteristics, whereas benchmark test assets involve only up to two characteristics.[18]

In summary, P-Tree factor models accurately price P-Tree test assets, while benchmark factors do not, and perform similarly or better than benchmark factor models in pricing benchmark test assets. P-Tree test assets set a high standard for testing asset pricing models, surpassing common sorted and industry portfolios.

4.4. P-Tree factors for investment

The P-Tree framework clusters thousands of assets into several leaf basis portfolios, reducing portfolio optimization complexity. These P-Tree test assets can be used as factors directly or as building blocks toward MVE portfolios for investment. Additionally, the frequency of rebalancing P-Tree investment strategies can be reduced to quarterly or annually to decrease transaction costs.

Full-sample evaluation. We evaluate the investment performance of P-Tree factors by combining them into one — the tangency portfolio of P-Tree factors. Table 7, Panel A, reports the P-Tree investment strategy's annualized Sharpe ratios and monthly alphas in the 40-year sample. The Sharpe ratio increases from 6.37 to 9.19, 11.21, 13.83, and 15.64 for the one-factor strategy to 5, 10, 15, and 20 factors, respectively. To assess their additional investment information, we further evaluate these P-Tree investment strategies over benchmark factor models. These model-adjusted alphas are all greater than 0.80% and highly statistically significant.

The existing research finds that P-Tree investment strategies have highly competitive Sharpe ratios. The correlation-based clustering in Ahn et al. (2009) generates basis portfolios that underperform bivariate-sorted portfolios on ME and BM. Furthermore, Daniel et al. (2020) nearly doubles the Sharpe ratio of the characteristics-sorted

[18] Industry portfolios are an exception because they are not sorted based on characteristics.

Table 6
Asset pricing performance: Cross-sectional R^2.
This table displays the cross-sectional R^2 (%) of pricing different sets of test assets by the P-Tree factor models and benchmark factor models. The factor models are listed in the columns, and the different specifications of test assets are listed in the rows. Specifically, the table includes 1, 5, 10, and 20 P-Tree factor models and FF5, Q5, RP-PCA, and IPCA five-factor models in the columns. In the rows, the test assets are from the top 1, 5, 10, and 20 P-Trees, univariate-sorted portfolios, bivariate-sorted portfolios, ME-BM 5 × 5 portfolios, and 49 industry portfolios.

	P-Tree1F	P-Tree5F	P-Tree10F	P-Tree20F	FF5	Q5	RP5	IP5
Panel A: 40 Years (1981–2020)								
P-Tree1	98.3	99.2	–	–	36.8	33.9	51.1	73.5
P-Tree1–5	65.0	82.6	90.2	95.9	54.3	55.9	56.8	49.6
P-Tree1–10	68.5	74.0	87.9	92.1	66.9	70.0	69.1	62.4
P-Tree1–20	72.1	77.9	84.3	88.9	73.8	76.8	77.8	68.6
Uni-Sort	92.7	97.3	98.5	98.7	97.0	97.9	98.0	95.3
Bi-Sort	88.7	97.2	98.0	98.5	96.1	97.5	97.3	92.9
ME-BM	88.4	98.6	98.8	99.4	96.8	97.4	97.0	96.6
Ind49	82.0	92.9	95.3	97.8	96.1	95.9	95.7	91.1
Panel B: 20 Years (1981–2000)								
P-Tree1	99.3	99.3	–	–	54.9	59.7	37.6	79.5
P-Tree1–5	49.2	88.5	92.6	96.0	25.8	31.8	41.1	48.6
P-Tree1–10	47.7	59.7	77.8	82.3	37.7	40.5	47.8	46.8
P-Tree1–20	42.8	54.0	70.2	74.5	42.2	45.6	52.5	50.2
Uni-Sort	80.4	96.5	96.9	97.6	94.6	95.9	97.0	94.5
Bi-Sort	66.4	92.1	93.9	94.9	89.1	91.5	92.7	91.1
ME-BM	66.6	94.0	96.2	99.0	93.4	95.3	96.4	94.5
Ind49	60.9	86.5	87.9	95.0	91.0	89.7	92.1	84.1
Panel C: 20 Year (2001–2020)								
P-Tree1	99.2	99.7	–	–	52.2	43.6	69.7	53.9
P-Tree1–5	71.4	87.6	92.3	94.3	60.0	60.7	64.9	61.3
P-Tree1–10	70.4	78.1	86.4	88.4	66.4	65.7	69.8	65.4
P-Tree1–20	70.5	76.4	79.8	82.8	71.7	69.9	75.4	65.2
Uni-Sort	87.5	96.6	97.6	97.9	97.3	97.2	97.4	95.7
Bi-Sort	93.0	97.2	97.9	98.4	97.9	97.5	98.0	96.9
ME-BM	92.9	95.7	98.6	99.3	97.5	97.5	97.4	96.1
Ind49	75.3	92.6	94.8	97.7	90.2	90.8	92.8	86.0

portfolio from 1.17 to 2.13 by constructing the corresponding characteristic-efficient portfolios. However, a single P-Tree delivers a 6.37 Sharpe ratio over 40 years from 1981 to 2020, representing a significant improvement over Ahn et al. (2009) and Daniel et al. (2020), although the empirical samples differ.

Out-of-sample evaluation. The results in Panel A of Table 7 are based on a full-sample analysis, which may raise concerns about in-sample overfitting. Therefore, we provide two exercises on OOS investment strategies. We follow Kozak et al. (2020) and perform a half-half split to construct the training and test samples. It suffices to demonstrate the OOS performance when the past and future samples predict each other. This approach further mitigates concerns regarding the look-ahead bias when models trained by the future sample predict the past.

On the one hand, the past-predicting-future results are reported in Panels B1 and B2. The first twenty-year in-sample Sharpe ratios are even higher than the forty-year ones in Panel A, and the OOS Sharpe ratios are over 3. Despite decreasing performance, many OOS alphas, adjusted by benchmark models, are still close to 1% and highly significant. On the other hand, for the future-predicting-past result, the OOS Sharpe ratios are larger than 3.87, and the model-adjusted alphas are over 0.80%. These findings demonstrate strong OOS evidence for P-Tree test assets and factors for spanning the MVE frontier.

The OOS investment performance results reported here are similar to those in recent studies. The deep reinforcement learning strategy in Cong et al. (2021) achieves a Sharpe ratio consistently above 2 from 1990 to 2021 even after excluding small stocks. Based on sorted portfolios on size, operating profitability, and investment in Bryzgalova et al. (2023), the regularized portfolio achieves an annualized Sharpe ratio of 2.39 from 2004 to 2016. The deep neural network approach in Feng et al. (2024) reports Sharpe ratios ranging from 2.95 to 3.00 from 2002 to 2021. P-Tree strategies stand out for their transparency, ease of computation, and flexibility as an all-around framework for test asset generation, asset pricing, and investment management.

5. Model extensions and discussion

We extend the P-Tree framework and applications along several directions, covering interpretability, model complexity, characteristic utility, and macroeconomic regimes.

5.1. Random P-Forest and P-Tree interpretability

Random forest (Breiman, 2001) grows multiple “decorrelated” trees on bootstrap training samples, which are random samples of observations with replacement and a subset of variables from the complete training set. This is called bagging, which helps mitigate model overfitting and quantify the uncertainty of model estimation. We can adopt this ensemble scheme and grow multiple “decorrelated” P-Trees on bootstrap training samples to form a random P-Forest. Consequently, there are two direct applications for the random P-Forest: assessing the characteristic importance and recovering a robust SDF using the large number of “decorrelated” P-Trees.

Characteristic importance. The simulation evidence in Internet Appendix I shows how the nonlinear and interactive P-Tree can recover the true set of characteristics, despite redundant or useless characteristics, leading to improved risk proxies for efficient frontier estimation. This contrasts with a linear factor model with ad hoc factors that might disregard these characteristics. The traditional sequential or independent asset sorting scheme usually focuses on up to three characteristics. Enumerating all possible sorting cases with multiple characteristics is NP-hard, but P-Tree’s iterative growth algorithm efficiently overcomes

Table 7
Factor investing by boosted P-Trees.
This table presents the investment performance of the initial factors produced by P-Tree and the MVE portfolio of multi-factors generated by boosted P-Trees. The reported statistics include the annualized Sharpe ratio and alphas with respect to CAPM, FF5, Q5, RP-PCA, and IPCA five-factor models. Panel A shows the 40-year sample from 1981 to 2020. Panels B1 and B2 present in- and out-of-sample results from 1981 to 2000 and 2001 to 2020, respectively. Panels C1 and C2 display in- and out-of-sample results from 2001 to 2020 and 1981 to 2000, respectively. We find all the α's are significant at 1% confidence level.

	SR	α_{CAPM}	α_{FF5}	α_{Q5}	α_{RP5}	α_{IP5}
Panel A: 40 Years (1981–2020)						
P-Tree1	6.37	1.39	1.37	1.36	1.28	1.12
P-Tree1–5	9.19	0.97	0.95	0.93	0.87	0.82
P-Tree1–10	11.21	1.01	1.00	0.98	0.93	0.89
P-Tree1–15	13.83	0.95	0.94	0.93	0.90	0.87
P-Tree1–20	15.64	0.97	0.96	0.95	0.93	0.90
Panel B1: 20 Years In-Sample (1981–2000)						
P-Tree1	7.13	1.86	1.78	1.72	1.62	1.59
P-Tree1–5	12.74	1.54	1.51	1.48	1.37	1.41
P-Tree1–10	19.22	1.51	1.49	1.49	1.43	1.43
P-Tree1–15	28.43	1.42	1.41	1.40	1.37	1.39
P-Tree1–20	38.01	1.36	1.35	1.34	1.32	1.34
Panel B2: 20 Years Out-of-Sample (2001–2020)						
P-Tree1	3.23	1.35	1.31	1.23	1.04	0.93
P-Tree1–5	3.41	1.02	1.00	0.95	0.77	0.62
P-Tree1–10	3.21	0.95	0.94	0.89	0.74	0.56
P-Tree1–15	3.12	0.89	0.89	0.83	0.69	0.48
P-Tree1–20	3.13	0.85	0.84	0.78	0.66	0.49
Panel C1: 20 Years In-Sample (2001–2020)						
P-Tree1	5.83	1.51	1.47	1.50	1.52	1.69
P-Tree1–5	9.32	1.30	1.29	1.28	1.30	1.31
P-Tree1–10	14.35	1.12	1.11	1.11	1.11	1.09
P-Tree1–15	20.64	1.08	1.07	1.08	1.10	1.05
P-Tree1–20	26.57	1.09	1.08	1.08	1.10	1.11
Panel C2: 20 Years Out-of-Sample (1981–2000)						
P-Tree1	4.35	1.50	1.42	1.35	1.60	1.58
P-Tree1–5	3.87	1.18	1.05	0.96	1.23	1.24
P-Tree1–10	4.29	1.02	0.93	0.85	1.14	1.10
P-Tree1–15	4.03	0.96	0.86	0.80	1.07	1.02
P-Tree1–20	3.88	0.96	0.87	0.81	1.08	1.03

this problem, making computation feasible.

The ensemble steps for constructing the random P-Forest are as follows: (i) Bootstrap the data on the time-series horizon with replacement and preserve the complete cross section of the panel data for the selected time periods to exploit the low serial correlations among asset returns. (ii) Randomly select 20 characteristics and independently grow P-Trees on each bootstrap sample. These two steps are repeated 1000 times to create a forest of 1000 P-Trees.

We study how often a characteristic is chosen for a split in the random P-Forest. A characteristic selected more often is seen as more important for maximizing mean–variance efficiency. In each bootstrap sample, 20 characteristics are randomly drawn to grow the tree, with only some chosen as split variables. We count the number of times a specific lth characteristic z_l is used in the first J splits and the total number of appearances. We define the measure of characteristic importance as follows:

$$\text{Select. Prob.}\left(z_l\right) = \frac{\#\left(z_l \text{ is selected at first } J \text{ splits }\right)}{\#\left(z_l \text{ appears in all bootstrap subsamples }\right)}. \tag{7}$$

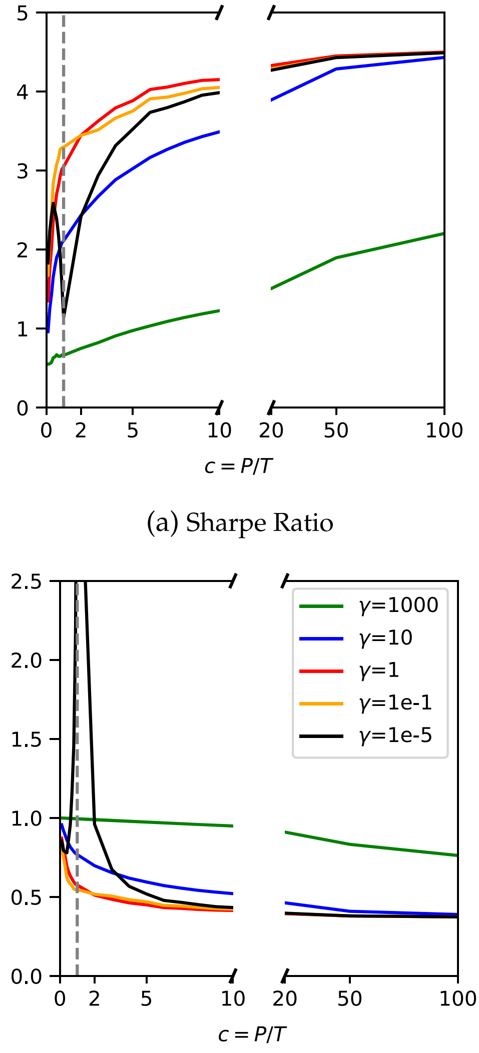
Table 8 summarizes the selection probabilities (7) of characteristics for $J = 1, 2, 3$. The selection probability increases as J grows from 1 to 3, so we must compare within the row. For each row, we list the top five selected characteristics.

The earnings surprise (SUE), one of the most important fundamentals, has a 50% chance of being the first splitting characteristic and a 66% chance of being one of the top two splitting characteristics in the bootstrap sample. While not incorporated into linear factor models like FF5 and Q5, SUE is most valuable in nonlinear interactive modeling for maximizing the mean–variance efficiency. Other frequently selected characteristics are return volatility (SVAR), dollar trading volume (DOLVOL), idiosyncratic volatility (RVAR_CAPM), change in profit margin (CHPM), and industry-adjusted book-to-market ratio (BM_IA). The characteristics in Table 8 cover four major categories – momentum, frictions, profitability, value – in the top splits. Note that characteristics not selected do not necessarily mean they are useless, as this table only shows the top three splits.

Table 8
Characteristic importance by selection probability.
This table reports the most frequently selected characteristics from the random P-Forest of 1000 trees. The "Top 1" rows only count the first split for 1000 trees. The "Top 2" or "Top 3" rows only count the first two or three splits. The numbers reported are the selection frequency for these top characteristics selected out of the 1000 ensembles. The descriptions of characteristics are listed in Table 1.

	1	2	3	4	5
Top1	SUE	SVAR	CHPM	RVAR_CAPM	BASPREAD
	0.51	0.34	0.25	0.23	0.21
Top2	SUE	SVAR	DOLVOL	CHPM	BM_IA
	0.66	0.38	0.34	0.31	0.31
Top3	SUE	DOLVOL	BM_IA	SVAR	ME_IA
	0.71	0.53	0.48	0.41	0.41

Interpretability V. S. overfitting. The random P-Forest may be over-parameterized and challenging for visualization and interpretation, but it offers several advantages. (i) Random P-Forest addresses model uncertainty and can evaluate characteristic importance by multiple bootstrap samples. In other words, the single P-Tree is readily interpretable as it visually depicts characteristics for non-linearities and



(a) Sharpe Ratio

(b) Pricing Error

Fig. 8. OOS Performance of Random P-Forest SDF.
This figure reports the Sharpe ratio, and pricing error (HJD) of the realized OOS SDF portfolio. Each P-Tree in the Random P-Forest is fed with 10 characteristics, randomly drawn without replacement from 61 ones, and generates 10 leaf basis portfolios. These P-Trees are independently trained. P-Trees are split with goal-oriented criteria to maximize the collective Sharpe ratio of leaf basis portfolios. The total number of leaf basis portfolios is denoted P. The horizontal axis shows model complexity $c = P/T$, with c ranging from 0.1 to 100 and $T = 240$ months. We report five specifications with shrinkage parameter γ in $[10^{-5}, 10^{-1}, 1, 10, 1000]$.

asymmetric interactions, but the model's efficacy may be limited if we cannot demonstrate that it does not overfit. In contrast, random P-Forest utilizes multiple random bootstrap samples, which helps mitigate the risk of overfitting. (ii) Random P-Forest offers an alternative to boosting for creating multiple test assets. Unlike boosting, which generates assets sequentially, random P-Forest independently creates assets across P-Trees.

In addition to the OOS evaluation in Table 7 for assessing model overfitting, we also compare key characteristics between the random P-Forest and the single P-Tree. The selected characteristics of the single P-Tree are also shown as important by the random P-Forest, suggesting that the single P-Tree is not overfitted. The consistent results may be attributed to the P-Tree's global split criteria, which prevent overfitting, similar to ML ensemble methods, while providing greater interpretability and transparent insights into non-linearities and asymmetric interactions. The significant characteristics obtained from the random P-Forest, such as earning surprise (SUE), dollar trading volume (DOLVOL), and industry-adjusted book-to-market ratio (BM_IA), are also chosen in the top splits of P-Tree for the 40 years, as illustrated in Fig. 4. In addition, the important characteristics listed in Table 8 are also selected in the top splits displayed in the P-Tree diagram for the subsample analysis (refer to Fig. A.1).

5.2. *Random P-Forest and complexity*

Our random P-Forest also relates to the growing popularity of large models in finance. In this recent literature, researchers usually question whether model performance improves as the model size increases, which joins the recent statistical literature on "Benign Overfitting" and "High-Dimensional Interpolation" (e.g., Belkin et al., 2019; Hastie et al., 2022) and "the Virtue of Complexity" in financial applications (e.g., Kelly et al., 2022, 2024; Didisheim et al., 2024). This literature shows that large regularized models, which have a large number of parameters relative to the number of observations, can also perform well out-of-sample, and their performance even improves as the number of parameters increases. This goes against the traditional wisdom of using parsimonious modeling in statistics and finance.

Random P-Forest constructs numerous uncorrelated P-Trees independently on randomly bootstrapped samples, allowing it to be expanded as a large model with both statistical and economic regularizations. These hundreds of P-Trees and thousands of leaf basis portfolios are similar to the randomly generated portfolios in Didisheim et al. (2024). We can estimate the SDF (tangency portfolio) using all the leaf basis portfolios in the forest and report their OOS performances to demonstrate the large model performance.

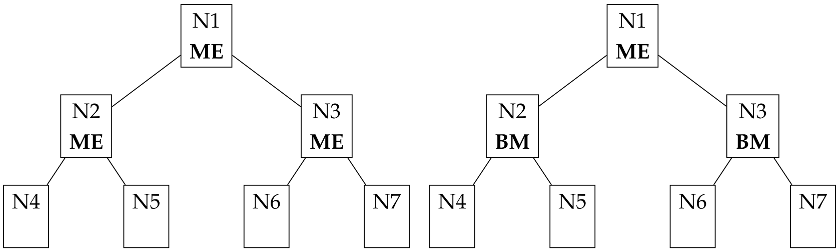
We create this random P-Forest SDF by (i) randomly selecting L characteristics for a P-Tree, growing P-Tree with split criteria (4). (ii) repeating the procedure (i) for B times to form a forest of B P-Trees with parallel computing. (iii) estimating the SDF weight on all the leaf basis portfolios in the random P-Forest with (8) based on in-sample data. We adopt the *ridge (ridgeless) SDF estimator* in Didisheim et al. (2024) to estimate the SDF weights $\hat{w}$ on leaf basis portfolios[19]:

$$\hat{w}(\gamma) = \arg\min_{w} E\left[\left(1 - w'\mathbf{R}_t\right)^2 + \gamma\|w\|^2\right], \tag{8}$$

where γ is a shrinkage parameter. We also investigate the large model performance and define $c = P/T$ as the degree of parameterization or *complexity*, where P is the number of leaf basis portfolios, and $T = 240$ is the fixed estimation rolling window. We examine the effects of extensive parameterization and shrinkage estimation on the random P-Forest SDF across various values for c and γ. In addition to the OOS Sharpe ratio, we also report the OOS Pricing Error $= E_{OOS}[(1 - \hat{w}'\mathbf{R}_t)^2]$. The Pricing Error is the OOS HJ distance (Hansen and Jagannathan, 1997), which has the above expression when $P > T_{OOS}$ and both are sufficiently large.

Fig. 8 displays the OOS performances of the random P-Forest SDF with $L = 10$ random characteristics from 2001 to 2020. As c increases, we find (i) the Sharpe ratio exhibits double ascent for low shrinkage cases and permanent ascent for high shrinkage cases, and (ii) the pricing error decreases for high shrinkage cases, with a spike around $c = 1$ for low shrinkage cases. These patterns are similar to those in Didisheim et al. (2024), demonstrating the benefit of randomness and large model size in the OOS performance of the random P-Forest SDF. For $c = 10$, the Sharpe ratio is about 4.0, and the pricing error is below 0.44. For $c = 100$, the Sharpe ratio becomes about 4.5, and the pricing error is below 0.38.

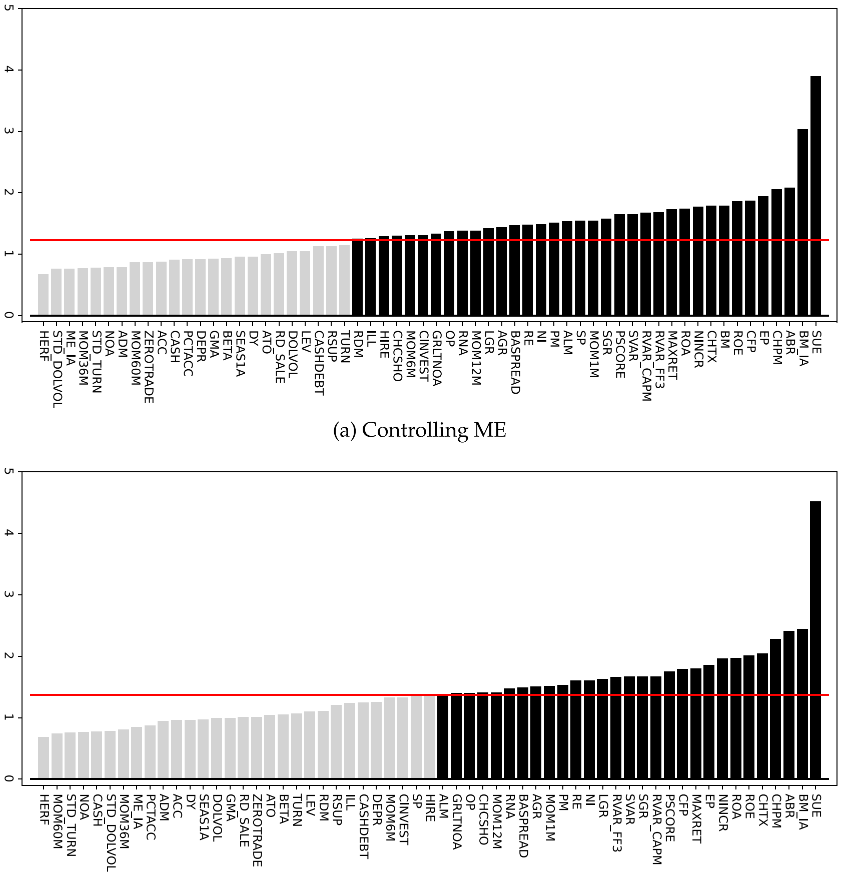
[19] Britten-Jones (1999) introduces a regression approach to estimate the tangency portfolio and Ao et al. (2019) extend the framework to the regularized portfolio by allowing a large number of assets.



(a) ME Baseline P-Tree

(b) ME-BM Baseline P-Tree

Fig. 9. Demonstration for ME baseline and ME-BM baseline P-Trees.



(a) Controlling ME

(b) Controlling ME-BM

Fig. 10. Evaluating a characteristic with P-Tree.
This figure shows the Sharpe ratio of P-Tree basis portfolios after splitting on a characteristic, based on either ME or ME-BM baseline P-Trees. Sub-figures (a) and (b) display the results for ME and ME-BM baseline P-Trees, respectively. We sort the characteristics by their Sharpe ratios in ascending order. The red lines show the Sharpe ratios of P-Tree basis portfolios after splitting one step further on the controlling characteristics. A bar above the red line indicates incremental information against the benchmark characteristics and is colored black. Otherwise, the gray bar indicates the characteristic does not provide incremental information. The characteristics are listed in Table 1.

Goal-oriented P-Tree v.s. random split P-Tree. The random P-Forest SDF involves randomness as a subset of characteristics is randomly drawn for each P-Tree, but split decisions are not made randomly. Goal-oriented search in P-Tree helps the leaf basis portfolios to maximize the collective Sharpe ratio efficiently. However, Didisheim et al. (2024) show that random portfolios can also generate the SDF with excellent OOS performances if the number of portfolios is large enough. We show the significance of goal-oriented search in maximizing mean–variance efficiency in an over-parameterized setting by presenting an alternative SDF specification for comparison. Specifically, the *random split SDF* uses random split criteria to create leaf basis portfolios, with no specific split criterion and characteristics and splitting values being randomly chosen for each decision to generate ten leaf basis portfolios.

The random P-Forest SDF exhibits comparable patterns to the random split SDF, but the random P-Forest is significantly more efficient regarding parameterization and computational expense. Specifically, it requires less complexity to achieve the same level of Sharpe ratio and pricing error as the random split SDF. Therefore, goal-oriented search improves the efficiency of large models with both statistical and economic regularizations, aligning with the idea of combining economic objectives with ML models in finance research. Internet Appendix III elaborates further.

Finally, from the test asset construction perspective, the goal-oriented P-Tree provides practically useful test assets for asset pricing model estimation and evaluation. The number of test assets is typically less than 100 because the statistical power of the asset pricing test is low when this number is large (Fan et al., 2015).

5.3. Evaluating a characteristic with P-Tree.

As an alternative pricing kernel, P-Tree can directly evaluate a new characteristic's incremental (and nonlinear) contribution over a benchmark to complement the literature. Researchers develop new factors by sorting based on specific characteristics in empirical studies and then test their independent information by controlling benchmark models through spanning regression. However, empirical studies rarely examine whether the associated characteristics provide new information and may cause information loss. To address this issue, P-Tree provides a new perspective for evaluating the usefulness of characteristics in generating test assets and latent factor models, given a set of benchmark characteristics, such as ME and BM.

In Fig. 9, we show two baseline P-Trees: a three-layer ME baseline P-Tree with all splits based on ME and a 3-layer ME-BM baseline P-Tree with the first layer split on ME and the second layer split on BM. To assess the incremental utility of estimating the efficient frontier of a characteristic against ME, we extend the ME baseline P-Tree with an additional split on the characteristic of interest or ME. One might expect a split to increase the Sharpe ratio of an MVE portfolio for a shallow P-Tree. Therefore, We only assess if a characteristic offers new information by comparing the Sharpe ratio improvement when adding it to the baseline P-Tree over a split on ME. This method helps us gauge the incremental impact of the characteristic in question.

In Fig. 10, subfigure (a) shows the incremental Sharpe ratio of each characteristic on the ME baseline P-Tree. We highlight characteristics outperforming ME in black and those underperforming ME in gray. We also sort the characteristics by their Sharpe ratios. Even for a shallow P-Tree, 24 characteristics do not provide incremental information against the ME baseline P-Tree. Further, we identify categories other than size, including momentum (SUE, ABR, NINCR), value (BM_IA, BM, CFP, EP), profitability (CHPM, ROA, ROE), and frictions/volatility (MAXRET, SVAR), that vastly improve investment performance by a simple further split on the ME baseline P-Tree.

In summary, we assess each characteristic on a benchmark P-Tree by controlling ME and/or BM for its marginal contribution to the investment objective. We can expect these marginal contributions to be smaller when considering a deeper tree as the benchmark. The relative importance of a characteristic may change when the benchmark P-Tree or controlling characteristics are altered, highlighting the adaptable nature of P-Tree evaluation. This evaluation can complement the conventional factor-spanning regression in evaluating characteristics.

Similarly, we assess each characteristic using the ME-BM baseline P-Tree shown in subfigure (b) of Fig. 9. This P-Tree has three layers and is based on ME and BM. With a stronger benchmark, some characteristics, such as RDM, ILL, HIRE, MOM6M, CINVEST, SP, do not provide additional information on this upgraded ME-BM baseline P-Tree. Against the ME-BM baseline P-Tree, we find that 30 characteristics do not offer incremental information for estimating the efficient frontier. Consistent with the ME baseline P-Tree results, momentum, value, and profitability characteristics are useful in the nonlinear interactive model.

5.4. P-Tree with macroeconomic regimes

Increasing empirical evidence indicates factor risk premia and loadings can vary significantly under various macroeconomic conditions. Given that the cross-sectional driving forces may change over time, examining how P-Tree performs in different states and which characteristics are selected for the mean–variance diversification is valuable. This subsection presents the subsample analysis for implementing P-Trees under various macro regimes and demonstrates their robust performance.

Characteristics evaluation under regimes. We conduct robustness checks by analyzing ten macro variables from Table 2 for subsample analysis in different regimes. The empirical results of P-Tree models under various regimes are shown in Table 9. We divide the sample into top and bottom regimes using the 50% 10-year rolling-window percentiles and fit a separate model for each regime. This allows us to assess the effectiveness of P-Tree in both positive and negative macroeconomic environments. [20]

First, the useful characteristics may vary depending on different macroeconomic conditions. P-Tree performs well, adjusts to macroeconomic changes, and chooses different characteristics for the top three splits. For example, when the term spread is low, or market volatility is high, the roles of SUE and DOVOL for firm fundamentals and trading volume are more important for cross-sectional mean–variance diversification. However, characteristics such as BASPREAD and RVAR_CAPM, representing frictions for liquidity and volatility, are often chosen for scenarios involving high net equity issuance, low default yield, low market liquidity, or low inflation.

Second, P-Tree test assets consistently provide informative results for asset pricing considerations: the GRS tests against Fama–French five factors are rejected for each regime, and the Sharpe ratios of the MVE portfolios spanned by P-Tree test assets are all greater than 7. Interestingly, the GRS statistics (weighted pricing errors) are significantly larger for scenarios that involve low inflation or a low t-bill rate, which implies the weakest moments of commonly used factors capturing the cross-sectional signals. The average alphas are larger than those of "P-Tree1" in Table 7, Panel A, implying the test assets under extreme macro regimes are even more challenging to price by common factors. Finally, we combine the P-Tree factors in two regimes and find the "Com" (Combined) factor has larger Sharpe ratios than the first P-Tree factor in Table 7, Panel A, which implements an unconditional model.

6. Conclusion

Estimating the mean–variance efficient frontier using individual asset returns and creating diversified test assets for asset pricing model evaluations constitute longstanding empirical challenges (e.g., Markowitz, 1952; Lewellen et al., 2010; Daniel et al., 2020). Our paper introduces a new class of tree-based models, Panel Tree (P-Tree), that effectively addresses these empirical challenges and analyzes the (unbalanced) panel of individual asset returns by generalizing multi-characteristics security sorting and splitting the cross section. Under the global criteria of mean–variance efficiency, P-Tree utilizes high-dimensional characteristics, which contain rich information on the joint distribution of asset returns, to generate characteristics-managed portfolios and recover the stochastic discount factor. More generally, P-Trees expand tree-based and other ML models beyond pure prediction, effectively analyzing panel data by economic-guided objectives, while maintaining interpretability and handling asymmetric and nonlinear interactions in low signal-to-noise environments.

Our empirical study of U.S. equities shows that P-Tree test assets significantly advance the efficient frontier compared to that spanned by those commonly used test assets (e.g., bivariate- or univariate-sorted portfolios) and exhibit significant unexplained alphas against benchmark models (e.g., Fama–French factors), highlighting the importance of test assets. These findings remain consistent and robust for the out-of-sample evaluation in the recent two decades. Second, the P-Tree tangency portfolio is constructed as traded factor models, outperforming popular observable and ML latent factor models for factor investment and cross-sectional pricing performance. Third, we demonstrate the versatility and utility of sparse tree-based models for economic interpretation. For example, we identify SUE, DOLVOL, and

[20] For identifying macro regime shifts through time-series splits, Feng et al. (2025) adapt the P-Tree framework for currency return dynamics.

Table 9
P-Tree performance under regime switches.
This table shows the performance of P-Tree models across various macroeconomic states. The time series of each macro variable is split into "Top" and "Btm" regimes using the 50% 10-year rolling-window percentiles. A simple P-Tree is then built for each regime, and similar performance statistics are reported in Table 5, along with the top three splitting characteristics. The rows labeled "Com" report the average return and Sharpe ratio of the combined P-Tree factor of the "Top" and "Btm" regimes.

	Regime	AVG	SR	GRS	p-value	$\overline{\lvert\alpha\rvert}$	$\sqrt{\overline{\alpha^2}}$	$\overline{R^2}$	Top three characteristics		
	Top	1.78	6.14	71.84	0.00	1.27	1.57	52.69	SUE	DOLVOL	CFP
DFY	Btm	1.21	4.56	40.01	0.00	1.08	1.44	60.05	ABR	RVAR_CAPM	STD_DOLVOL
	Com	1.49	5.16								
	Top	1.45	5.84	43.17	0.00	1.03	1.30	73.16	CHPM	BM_IA	ME
DY	Btm	1.30	6.21	101.56	0.00	1.05	1.21	51.45	ABR	ZEROTRADE	DOLVOL
	Com	1.35	6.03								
	Top	0.99	5.51	34.85	0.00	0.94	1.28	78.21	RVAR_CAPM	ME_IA	ME
EP	Btm	1.34	6.04	98.62	0.00	0.94	1.18	56.89	SUE	DOLVOL	BM_IA
	Com	1.24	5.76								
	Top	1.10	5.90	89.15	0.00	0.81	1.04	62.31	RVAR_CAPM	SUE	STD_DOLVOL
ILL	Btm	1.50	7.44	72.13	0.00	1.03	1.30	59.38	SUE	DOLVOL	SP
	Com	1.24	6.21								
	Top	1.16	4.83	30.10	0.00	0.99	1.29	75.20	CFP	ATO	ME
INFL	Btm	1.29	6.61	107.40	0.00	0.93	1.26	57.67	SUE	DOLVOL	CFP
	Com	1.25	5.89								
	Top	1.33	5.10	38.77	0.00	0.89	1.14	65.03	CFP	ALM	STD_DOLVOL
LEV	Btm	1.39	5.36	66.94	0.00	1.23	1.46	52.23	NI	MOM12M	BM_IA
	Com	1.37	5.26								
	Top	1.37	5.05	45.70	0.00	1.67	2.16	53.03	RVAR_CAPM	ME_IA	RD_SALE
NI	Btm	1.51	6.71	93.02	0.00	1.18	1.38	61.79	ROA	DOLVOL	SUE
	Com	1.45	5.81								
	Top	1.24	6.22	72.03	0.00	0.94	1.24	62.66	SUE	DOLVOL	CFP
SVAR	Btm	1.23	4.78	43.92	0.00	1.18	1.38	59.41	ROA	ZEROTRADE	ABR
	Com	1.23	5.36								
	Top	1.09	7.15	59.53	0.00	0.75	0.91	61.98	CHCSHO	SUE	DOLVOL
TBL	Btm	1.30	7.05	134.30	0.00	0.93	1.10	63.44	SUE	DOLVOL	CFP
	Com	1.24	6.97								
	Top	1.17	6.01	67.58	0.00	0.99	1.28	63.45	EP	STD_DOLVOL	SUE
TMS	Btm	1.23	5.82	70.00	0.00	0.86	1.17	52.52	SUE	DOLVOL	BM_IA
	Com	1.20	5.91								

BM_IA as key characteristics that interact to explain the cross-section of asset returns. Finally, P-Tree captures the complexity of panel stock returns with sparsity, achieving exceptional OOS Sharpe ratios close to those of over-parameterized large models. Beyond asset pricing, our framework offers an interpretable and computationally efficient alternative to deep-learning-based AI for goal-oriented search in large modeling spaces.

CRediT authorship contribution statement

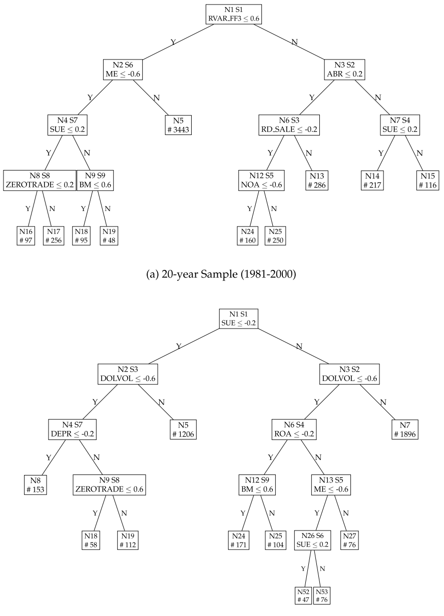
Lin William Cong: Writing – review & editing, Writing – original draft, Supervision, Resources, Methodology, Formal analysis, Conceptualization. **Guanhao Feng:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Jingyu He:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Xin He:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation.

Declaration of competing interest

None.

Appendix A

See Algorithm 1 and Fig. A.1.



(a) 20-year Sample (1981-2000)

(b) 20-year Sample (2001-2020)

Fig. A.1. Panel tree diagram for subsamples.
Format follows Fig. 4. Subfigure (a) shows P-Tree in the sample from 1981–2000. Subfigure (b) shows P-Tree in the sample from 2001–2020.

Algorithm Growing a Single P-Tree

```
procedure GrowTree(root)
Input Asset returns $\mathbf{r}_{i,t}$ and ranked characteristics $\mathbf{z}_{i,t}$
outcome Grow P-Tree from the root node, form leaf basis portfolios
    for $j$ from 1 to $J$ do                          ▷ Loop over number of iterations
        if current depth $\geq d_{\max}$ then
            return.
        else
            Search the tree, find all leaf nodes $\mathcal{N}$
            for each leaf node $N$ in $\mathcal{N}$ do        ▷ Loop over all current leaf nodes
                for each candidate split $\tilde{\mathbf{c}}_{k,m,N}$ in $\mathcal{C}_N$ do
                    Partition data temporally in $N$ according to $\tilde{\mathbf{c}}_{k,m,N}$.
                    if Either left or right child of $N$ does not satisfy minimal leaf size then
                        $\mathcal{L}(\tilde{\mathbf{c}}_{k,m,N}) = -\infty$.
                    else
                        Calculate leaf basis portfolios.
                        Estimate tangency portfolio using *all* leaf basis portfolios as in (2).
                        Calculate the split criteria $\mathcal{L}(\tilde{\mathbf{c}}_{k,m,N})$ in (4).
                    end if
                end for
            end for
            Find the best leaf node and split rule that maximize split criteria
                $\tilde{\mathbf{c}}_j = \underset{N\in\mathcal{N},\tilde{\mathbf{c}}_{k,m,N}\in\mathcal{C}_N}{\operatorname{argmax}} \{\mathcal{L}(\tilde{\mathbf{c}}_{k,m,N})\}$
            Split the node selected with the $j$-th split rule of the tree $\tilde{\mathbf{c}}_j$.
        end if
    end for
    return $\left\{\mathbf{R}_t^{(J)}, f_t^{(J)}\right\}$
end procedure
```

References

Ahn, D.-H., Conrad, J., Dittmar, R.F., 2009. Basis assets. Rev. Financ. Stud. 22 (12), 5133–5174.

Ait-Sahalia, Y., Xiu, D., 2017. Using principal component analysis to estimate a high dimensional factor model with high-frequency data. J. Econometrics 201 (2), 384–399.

Ang, A., Liu, J., Schwarz, K., 2020. Using Stocks or Portfolios in Tests of Factor Models. J. Financ. Quant. Anal. 55, 709–750.

Ao, M., Yingying, L., Zheng, X., 2019. Approaching mean-variance efficiency for large portfolios. Rev. Financ. Stud. 32 (7), 2890–2919.

Avramov, D., Cheng, S., Metzker, L., 2023. Machine learning vs. economic restrictions: Evidence from stock return predictability. Manag. Sci. 69 (5), 2587–2619.

Bali, T.G., Beckmeyer, H., Moerke, M., Weigert, F., 2023. Option return predictability with machine learning and big data. Rev. Financ. Stud. 36 (9), 3548–3602.

Banz, R.W., 1981. The relationship between return and market value of common stocks. J. Financ. Econ. 9 (1), 3–18.

Barillas, F., Shanken, J., 2017. Which alpha? Rev. Financ. Stud. 30 (4), 1316–1338.

Belkin, M., Hsu, D., Ma, S., Mandal, S., 2019. Reconciling modern machine-learning practice and the classical bias–variance trade-off. Proc. Natl. Acad. Sci. 116 (32), 15849–15854.

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

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

Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J., 1984. Classification and Regression Trees. Routledge.

Britten-Jones, M., 1999. The sampling error in estimates of mean-variance efficient portfolio weights. J. Financ. 54 (2), 655–671.

Bryzgalova, S., Pelger, M., Zhu, J., 2023. Forest through the trees: Building cross-sections of stock returns. J. Financ. Forthcom..

Capponi, A., Yu, S., 2024. Price Discovery in the Machine Learning Age. Technical report, Columbia University.

Chen, Y., Cliff, M., Zhao, H., 2017. Hedge funds: The good, the bad, and the lucky. J. Financ. Quant. Anal. 52 (3), 1081–1109.

Chen, L., Pelger, M., Zhu, J., 2024. Deep learning in asset pricing. Manag. Sci. 70 (2), 714–750.

Chipman, H.A., George, E.I., McCulloch, R.E., 2010. BART: Bayesian additive regression trees. Ann. Appl. Stat. 4, 266–298.

Chordia, T., Subrahmanyam, A., Anshuman, V.R., 2001. Trading activity and expected stock returns. J. Financ. Econ. 59 (1), 3–32.

Cochrane, J.H., 2011. Presidential address: Discount rates. J. Financ. 66 (4), 1047–1108.

Cong, L.W., Feng, G., He, J., Li, J., 2023. Uncommon Factors and Asset Heterogeneity in the Cross Section and Time Series. Technical report, City University of Hong Kong.

Cong, L.W., Feng, G., He, J., Wang, Y., 2024. Mosaics of Predictability. Technical report, City University of Hong Kong.

Cong, L.W., Tang, K., Wang, J., Zhang, Y., 2021. AlphaPortfolio: Direct Construction Trough Deep Reinforcement Learning and Interpretable AI. Technical report, Cornell University.

Creal, D., Kim, J., 2023. Empirical Asset Pricing with Bayesian Regression Trees. Technical report, University of Notre Dame.

Da, Z., Gurun, U.G., Warachka, M., 2014. Frog in the pan: Continuous information and momentum. Rev. Financ. Stud. 27 (7), 2171–2218.

Daniel, K., Grinblatt, M., Titman, S., Wermers, R., 1997. Measuring mutual fund performance with characteristic-based benchmarks. J. Financ. 52 (3), 1035–1058.

Daniel, K., Mota, L., Rottke, S., Santos, T., 2020. The cross-section of risk and returns. Rev. Financ. Stud. 33 (5), 1927–1979.

Didisheim, A., Ke, S., Kelly, B., Malamud, S., 2024. APT or "AIPT"? The Surprising Dominance of Large Factor Models. Technical report, Yale University.

Fama, E.F., French, K.R., 1996. Multifactor explanations of asset pricing anomalies. J. Financ. 51 (1), 55–84.

Fama, E.F., French, K.R., 2015. A five-factor asset pricing model. J. Financ. Econ. 116 (1), 1–22.

Fan, J., Ke, Z.T., Liao, Y., Neuhierl, A., 2022. Structural Deep Learning in Conditional Asset Pricing. Technical report, Princeton University.

Fan, J., Liao, Y., Yao, J., 2015. Power enhancement in high-dimensional cross-sectional tests. Econometrica 83 (4), 1497–1541.

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

Feng, G., He, J., Li, J., Sarno, L., Zhang, Q., 2025. Currency Return Dynamics: What Is the Role of U.S. Macroeconomic Regimes?. Technical report, City University of Hong Kong.

Feng, G., He, J., Polson, N.G., Xu, J., 2024. Deep Learning in Characteristics-Sorted Factor Models. J. Financ. Quant. Anal. 59 (7), 3001–3036.

Feng, G., Jiang, L., Li, J., Song, Y., 2023. Deep Tangency Portfolio. Technical report, City University of Hong Kong.

Freund, Y., Schapire, R.E., 1997. A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. System Sci. 55 (1), 119–139.

Gagliardini, P., Ossola, E., Scaillet, O., 2016. Time-varying risk premium in large cross-sectional equity data sets. Econometrica 84 (3), 985–1046.

Gibbons, M.R., Ross, S.A., Shanken, J., 1989. A test of the efficiency of a given portfolio. Econometrica 1121–1152.

Giglio, S., Xiu, D., Zhang, D., 2023. Test Assets and Weak Factors. J. Financ. Forthcom..

Gu, S., Kelly, B., Xiu, D., 2020. Empirical asset pricing via machine learning. Rev. Financ. Stud. 33 (5), 2223–2273.

Gu, S., Kelly, B., Xiu, D., 2021. Autoencoder asset pricing models. J. Econometrics 222 (1), 429–450.

Hansen, L.P., Jagannathan, R., 1997. Assessing specification errors in stochastic discount factor models. J. Financ. 52 (2), 557–590.

Hastie, T., Montanari, A., Rosset, S., Tibshirani, R.J., 2022. Surprises in high-dimensional ridgeless least squares interpolation. Ann. Stat. 50 (2), 949.

He, J., Hahn, P.R., 2023. Stochastic tree ensembles for regularized nonlinear regression. J. Amer. Statist. Assoc. 118 (541), 551–570.

He, A., Huang, D., Li, J., Zhou, G., 2023. Shrinking factor dimension: A reduced-rank approach. Manag. Sci. 69 (9), 5501–5522.

He, J., Yalov, S., Hahn, P.R., 2019. XBART: Accelerated Bayesian additive regression trees. In: The 22nd International Conference on Artificial Intelligence and Statistics. pp. 1130–1138.

Hoberg, G., Welch, I., 2009. Optimized vs. Sort-Based Portfolios. Technical report, University of Southern California.

Hou, K., Mo, H., Xue, C., Zhang, L., 2021. An augmented q-factor model with expected growth. Rev. Financ. 25 (1), 1–41.

Jarrow, R.A., Murataj, R., Wells, M.T., Zhu, L., 2023. The low-volatility anomaly and the adaptive multi-factor model. Int. J. Theor. Appl. Finance 26 (04n05), 2350020.

Jensen, T.I., Kelly, B.T., Malamud, S., Pedersen, L.H., 2024. Machine learning and the implementable efficient frontier. Rev. Financ. Stud. Forthcom..

Kelly, B., Malamud, S., Zhou, K., 2022. The virtue of complexity everywhere. Yale Universtiy.

Kelly, B., Malamud, S., Zhou, K., 2024. The virtue of complexity in return prediction. J. Financ. (1), 459–503.

Kelly, B., Pruitt, S., Su, Y., 2019. Characteristics are covariances: A unified model of risk and return. J. Financ. Econ. 134 (3), 501–524.

Kim, S., Korajczyk, R.A., Neuhierl, A., 2021. Arbitrage portfolios. Rev. Financ. Stud. 34 (6), 2813–2856.

Kozak, S., Nagel, S., 2022. When Do Cross-Sectional Asset Pricing Factors Span the Stochastic Discount Factor?. Technical report, University of Michigan.

Kozak, S., Nagel, S., Santosh, S., 2018. Interpreting factor models. J. Financ. 73 (3), 1183–1223.

Kozak, S., Nagel, S., Santosh, S., 2020. Shrinking the cross-section. J. Financ. Econ. 135 (2), 271–292.

Lee, C.M., Swaminathan, B., 2000. Price momentum and trading volume. J. Financ. 55 (5), 2017–2069.

Lettau, M., Pelger, M., 2020. Factors that fit the time series and cross-section of stock returns. Rev. Financ. Stud. 33 (5), 2274–2325.

Lewellen, J., Nagel, S., Shanken, J., 2010. A skeptical appraisal of asset pricing tests. J. Financ. Econ. 96 (2), 175–194.

Liu, W., 2006. A liquidity-augmented capital asset pricing model. J. Financ. Econ. 82 (3), 631–671.

Lopez-Lira, A., Roussanov, N.L., 2020. Do Common Factors Really Explain the Cross-Section of Stock Returns?. Technical report, University of Pennsylvania.

Markowitz, H., 1952. Portfolio selection. J. Financ. 7, 77–99.

Patton, A.J., Weller, B.M., 2022. Risk price variation: The missing half of empirical asset pricing. Rev. Financ. Stud. 35 (11), 5127–5184.

Pesaran, M.H., Yamagata, T., 2023. Testing for alpha in linear factor pricing models with a large number of securities. J. Financ. Econ..

Rendleman, Jr., R.J., Jones, C.P., Latane, H.A., 1982. Empirical anomalies based on unexpected earnings and the importance of risk adjustments. J. Financ. Econ. 10 (3), 269–287.

Rossi, A.G., Timmermann, A., 2015. Modeling covariance risk in Merton’s ICAPM. Rev. Financ. Stud. 28 (5), 1428–1461.

Sorensen, E.H., Miller, K.L., Ooi, C.K., 2000. The decision tree approach to stock selection. J. Portf. Manag. 27 (1), 42–52.