s:fi
RESEARCH

Swiss Finance Institute
Research Paper Series
N°22-63

Machine Learning and the Implementable Efficient Frontier

Theis Ingerslev Jensen
Yale School of Management

Bryan Kelly
Yale School of Management, AQR Capital Management, and NBER

Semyon Malamud
Ecole Polytechnique Fédérale de Lausanne, Swiss Finance Institute, and CEPR

Lasse Heje Pedersen
Copenhagen Business School, AQR Capital Management, and CEPR

Machine Learning and the Implementable Efficient Frontier

Theis Ingerslev Jensen, Bryan Kelly, Semyon Malamud, and Lasse Heje Pedersen*

This version: June 19, 2024

Abstract

We propose that investment strategies should be evaluated based on their net-of-trading-cost return for each level of risk, which we term the "implementable efficient frontier." While numerous studies use machine learning return forecasts to generate portfolios, their agnosticism toward trading costs leads to excessive reliance on fleeting small-scale characteristics, resulting in poor net returns. We develop a framework that produces a superior frontier by integrating trading-cost-aware portfolio optimization with machine learning. The superior net-of-cost performance is achieved by learning directly about portfolio weights using an economic objective. Further, our model gives rise to a new measure of "economic feature importance."

Keywords: asset pricing, machine learning, transaction costs, economic significance, investments
JEL Codes: C5, C61, G00, G11, G12

*Jensen is at Yale School of Management; www.tijensen.com. Kelly is at Yale School of Management, AQR Capital Management, and NBER; www.bryankellyacademic.org. Malamud is at Swiss Finance Institute, EPFL, and CEPR, and is a consultant to AQR. Pedersen is at AQR Capital Management, Copenhagen Business School, and CEPR; www.lhpedersen.com. We are grateful for helpful comments from Cliff Asness, Jules van Binsbergen, Darrell Duffie, Marc Eskildsen, Markus Ibert, Leonid Spesivtsev, and conference and seminar participants at American Finance Association 2024, Red Rock Finance Conference 2023, Oxford SMLFin online seminar, Copenhagen Business School and NBER Big Data and Securities Markets Conference, Fall 2021. AQR Capital Management is a global investment management firm that may or may not apply similar investment techniques or methods of analysis as described herein. The views expressed here are those of the authors and not necessarily those of AQR. Semyon Malamud gratefully acknowledges the financial support of the Swiss Finance Institute and the Swiss National Science Foundation, Grant 100018_192692. All four authors appreciate the financial support of INQUIRE Europe.

This paper studies how security information can be used for portfolio selection in a flexible and realistic setting with transaction costs. The goal is thus both to provide a powerful tool for portfolio choice and to shed new light on which security characteristics are economically important drivers of asset pricing.

The financial machine learning (ML) literature provides a flexible framework to combine several characteristics into a single measure of overall expected returns (e.g. Gu et al., 2020). The same literature documents the relative "feature importance" of different return prediction characteristics (e.g. Chen et al., 2023). These findings suggest that the prediction success of ML methods is often driven by short-lived characteristics that work well for small and illiquid stocks (e.g. Avramov et al., 2023), suggesting that they might be less critical for the real economy (e.g. Van Binsbergen and Opp, 2019). The high transaction costs of portfolio strategies based on ML imply that these strategies are difficult to implement in practice and, more broadly, raise questions about the relevance and interpretation of the predictability documented in this literature. Do ML-based expected return estimates merely tell us about mispricings that investors don't bother to arbitrage away because the costs are too large, the mispricing too fleeting, and the relevant stocks too small to matter? Or, do trading-cost-aware ML-based predictions also work for large stocks over significant periods and in a valuable way for large investors, thus providing information about their preferences?

This paper seeks to generate economically useful predictions. We are interested in deriving ML-driven portfolios that can be realistically implemented by important market participants such as large pension funds or other professional asset managers. If a strategy is implementable at scale, then the predictive variables that drive such portfolio demands are also informative about the equilibrium discount rates of major companies (Koijen and Yogo (2019)).

While ML with transaction costs is challenging to attack with brute force, we deliver a tractable solution through the help of economic theory. Specifically, we show how to integrate the ML problem into a generalized version of the optimal portfolio selection framework of Gârleanu and Pedersen (2013). The main thrust of our approach is to feed the objective function explicit knowledge of implementability, so it knows to search for perhaps subtle but "usable" predictive patterns while discarding more prominent but costly predictive patterns. We develop an ML method to produce optimal portfolios while considering realistic frictions from transaction costs of the securities it trades. Our solution also gives rise to a new

measure of "economic feature importance" that captures which characteristics provide the most investment value; in other words, which characteristics contribute the most to the overall portfolio's risk-adjusted returns after trading costs.

Our approach generalizes Gârleanu and Pedersen (2013) in three important ways. First, while Gârleanu and Pedersen (2013) assume that expected price changes are linear functions of a set of signals, we allow expected returns to be a fully general function of the signals, opening the door for flexible non-linear ML. Second, our setting is based on stationary returns, not stationary price changes, solving a vexing problem in the portfolio choice literature and providing a new coherence to empirical analysis over long horizons.[1] Third, while Gârleanu and Pedersen (2013) take the data generating process as given, we integrate the estimation process into the optimization process via ML, showing our method's practical and empirical power.

To understand the difference between our approach and the typical use of ML in finance, note that the latter takes a two-step approach: First, find a function of characteristics that predicts gross returns, and second, use the resulting forecasts to build portfolios. This typical approach abstracts from transaction costs and turnover, and the resultant investment strategies produce negative returns net of transaction costs.

Our approach builds transaction costs directly into the objective function, thus ensuring that the algorithm learns about usable predictability. One element of usable predictability is that it is relevant for large stocks with low transaction costs. Another essential element is alpha decay, which is how persistent a predictor is. With transaction costs, you will likely own whatever you buy today for a while because the trading costs encourage you to enter or exit positions only slowly. Naturally, understanding the expected return both now and further into the future is relevant.

Empirically, the optimal ML predictor of near-term returns is indeed different from the optimal ML predictor of returns far into the future. In other words, if f_h is the function that best predicts returns h months into the future, $E_t[r_{i,t+h}] = f_h(s_{it})$, then this function is

[1] Gârleanu and Pedersen (2013) show that the portfolio problem simplifies by looking at numbers of shares and price changes because this sidesteps the issue of portfolio growth that has plagued the literature. The portfolio growth is the issue that, if you put 10% of your wealth in IBM stock today, then you will not have 10% of your wealth in IBM next period before trading – because of the price change of IBM and other stocks. Working with the number of shares sidesteps this issue (the number of shares only changes when you trade). Still, the cost is that profit is equal to shares times price changes, so the model cannot be specified in terms of percentage returns, making empirical analysis difficult. We have found a way to work with empirically relevant units and preserve tractability via an approximately optimal solution.

different across h. Given that the standard ML approach uses only f_1, we see that it misses the information contained in f_h at other horizons, $h > 1$.

One way to implement our approach is to forecast returns across many time horizons $h = 1, 2, ...$, then to use all of the predictive functions, f_h, appropriately discounted given risk, risk aversion, and the form of the trading cost function. However, this approach requires a highly complex and computationally intensive ML formulation to accommodate all predictive functions simultaneously.

Our preferred approach is to learn directly about portfolio weights instead of expected returns.[2] This simple approach delivers an essentially closed-form solution to the highly complex portfolio problem in a single step!

To evaluate the performance of our method, we propose that portfolio choice methods and ML predictions are assessed based on the net-of-cost investment opportunities they produce. Indeed, a fundamental insight in portfolio choice is that investors can depict their investment opportunity set as all the achievable combinations of risk and expected return, giving rise to the "efficient frontier" depicted in most finance textbooks as the tangency line to the hyperbola generated by risky investments. The textbook frontier is drawn in a frictionless setting that abstracts from trading costs, but real-world investors care about their net return. What does the frontier look like when we take trading costs into account?

Panel A of Figure 1 illustrates frontiers for various methods that we study. The baseline for comparison is the cost-agnostic Markowitz-ML solution and the hyperbola of risky investments – both in *gross* terms– that is, our implementation of the textbook frontier using ML. Specifically, these portfolios use ML to build stock-level expected returns, use the academic analog of Barra to build a covariance matrix, and then form ex-ante efficient portfolios from these two inputs. Figure 1.A plots the portfolios' realized out-of-sample performance. As seen in the figure, while the Markowitz solution is tangent to the hyperbola in a textbook analysis with known means and variances, the Markowitz solution is not precisely tangent in our out-of-sample analysis. In any event, the Markowitz portfolio performs very well out-of-sample, delivering a Sharpe ratio of roughly 2.1 per annum. But this is in *gross* terms. The portfolio's turnover is enormous and the textbook frontier is non-implementable in practice.

The other lines in Figure 1 show our concept of an "implementable efficient frontier,"

[2]This idea builds on Brandt et al. (2009) who propose using portfolio weights that are parametric (linear) functions of characteristics in a setting without trading costs. We extend their idea to handle the much more complex portfolio problem with transaction costs and by using ML to learn about the optimal weights.

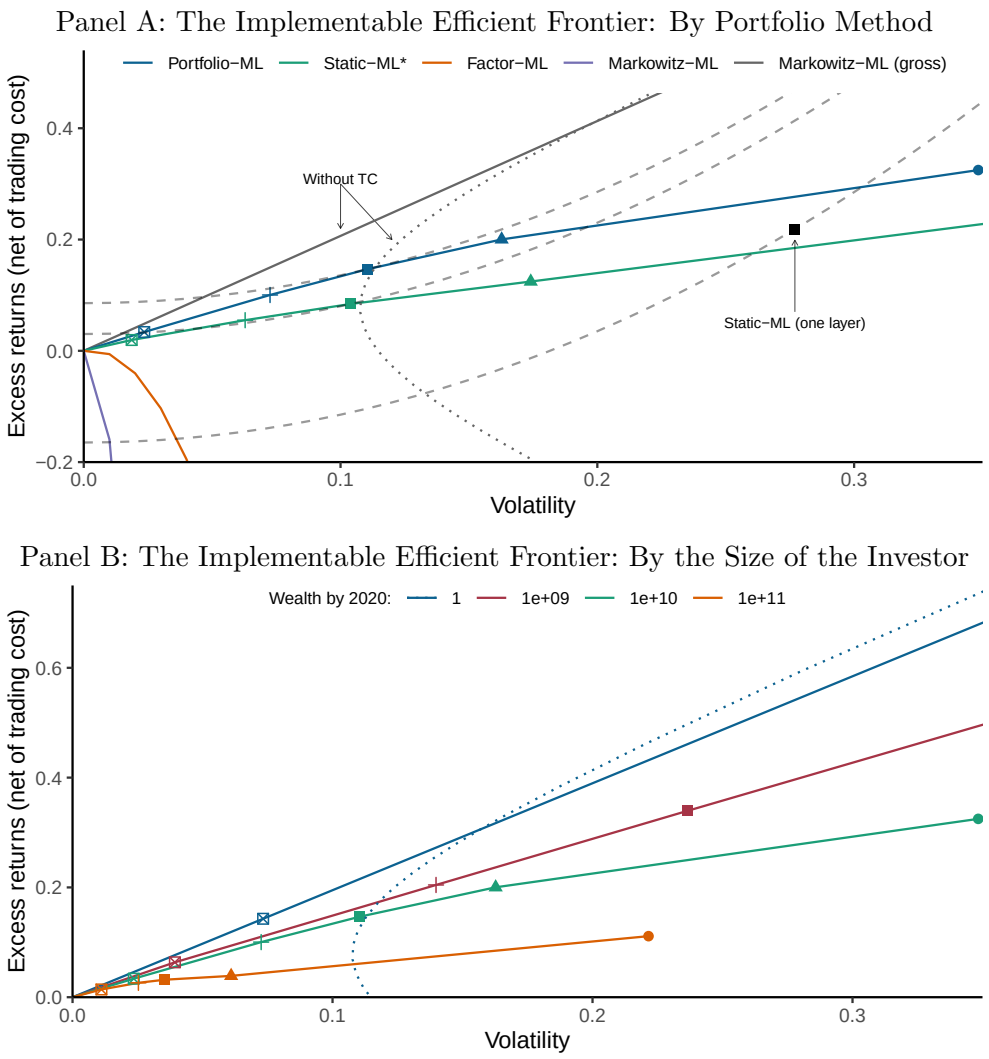

Figure 1: The Implementable Efficient Frontier: Risk vs. Return Net of Trading Costs

Note: Panel A shows the implementable efficient frontier for different portfolio methods with a wealth of \$10B by 2020. The dashed lines show indifference curves. The dotted hyperbola is the mean-variance frontier of risky assets without trading costs, implemented by estimating risk and expected return separately, out-of-sample. The grey line is the Markowitz-ML efficient frontier before trading cost. The grey line would be the tangency line to the hyperbola in a standard in-sample textbook analysis, but it is not exactly tangent out-of-sample. After trading costs, Markowitz-ML and Factor-ML have downward bending frontiers as these methods are not implementable. Static-ML produces a positive net Sharpe but negative utility, but it works well with an extra tuning layer, denoted Static-ML*. Our Portfolio-ML works significantly better out-of-sample. Panel B shows the implementable efficient frontier at different wealth levels. The dotted hyperbola is the same mean-variance frontier as in Panel A. The blue line is the optimal portfolio of risky and risk-free assets for an investor with zero wealth, corresponding to no trading costs, estimated using our Portfolio-ML method for different relative risk aversions. The lower lines illustrate the mean-variance frontiers with larger wealth levels, which are also estimated using Portfolio-ML. In both panels, the relative risk aversions are 1 (circle), 5 (triangle), 10 (square), 20 (plus), and 100 (boxed cross), and the sample period is 1981-2020. Further details are provided in Section 6.2.

that is, the achievable combinations of risk and expected return, net of trading costs. Focusing first on the Markowitz portfolio, we see that its net-of-cost, implementable frontier immediately dives into negative expected return territory as soon as it moves away from a 100% risk-free allocation, as seen in the bottom curve in Panel A. The shape of the implementable efficient frontier may be surprising: Whereas the textbook frontier is a straight line when increasing the allocation to the risky securities while reducing the risk-free allocation (or applying leverage), the true implementable frontier bends down because larger positions incur higher transaction costs. Said differently, we show that the net-of-cost Sharpe ratio declines along the implementable efficient frontier.

To understand the source of the problem for Markowitz-ML at a deeper level, the feature importance of this portfolio reveals the culprit: excessive reliance on fleeting small-scale characteristics (e.g., 1-month reversal for small stocks), which bear high turnover, high trading costs, and result in poor net returns. Further, Panel A of Figure 1 also shows that a standard portfolio sort based on the predicted returns, what we call "Factor-ML", is also not implementable.

The difficulty of the standard portfolios from the literature is noteworthy. Still, it is also interesting to compare our approach to a more sophisticated alternative that may be used by some large investors. This sophisticated alternative first uses ML to build expected returns (agnostic of trading costs), then in an additional second-stage optimization, takes transaction costs into account to build portfolios. This "Static-ML" approach delivers a positive net Sharpe but a lower utility than putting all the money in the risk-free asset, as seen from the indifference curves in Figure 1.A.

To create a more difficult benchmark to beat for our preferred method (Portfolio-ML), we further enhance the standard approach by adding several extra hyper-parameters that improve performance by adjusting its scale in various ways. We refer to this approach as Static-ML*, where the "*" indicates that we use an extra tuning stage. Static-ML* performs well, delivering high utility as seen in Figure 1.

Despite that Static-ML* is a sophisticated multi-stage approach that is much more highly parameterized than our Portfolio-ML method, our Portfolio-ML method nevertheless significantly outperforms Static-ML*. To understand why Static-ML* underperforms our approach, note that the first-stage ML procedure produces expected returns dominated by short-term signals. This method does not consider which predictors are persistent and which have quick

alpha decay. The second-stage optimization reduces turnover, especially for small stocks, which leads to a much better performance than portfolios that ignore transaction costs. However, this static approach can nevertheless be improved by recognizing the dynamic nature of the portfolio using a method that is sensitive to how expected returns vary across several return horizons (i.e., alpha decay).

In other words, our Portfolio-ML method delivers out-of-sample net-of-cost returns that outperform a highly sophisticated alternative by roughly 60% in Sharpe ratio terms and 290% in utility terms. Further, the feature importance across signals changes when we consider transaction costs. While short-term reversal signals highly influence naive methods, our method seeks to optimally blend return predictability across multiple future horizons, especially for liquid stocks. This leads to value and quality, earning the highest feature importance.

Panel B of Figure 1 draws the implementable efficient frontier using our Portfolio-ML at different levels of wealth or asset under management (AUM). Interestingly, while the textbook efficient frontier is the same for all investors, the implementable efficient frontier depends on the investor's size via the implied trading costs. Indeed, larger investors face worse (i.e., lower) efficient frontiers that "cut into" the hyperbola.

As an interesting benchmark, the top line shows the Portfolio-ML strategy when trading costs are nearly zero since the investor has an AUM near zero. This implementable frontier is obviously good due to the near-zero trading costs, but we note that such sophisticated ML-based trading is hardly feasible for small investors in the real world.[3]

The frontier at each wealth level shows that the set of optimal implementable portfolios is strictly worse for higher AUM investors. This degradation happens for two reasons. First, trading a larger portfolio incurs higher market impact costs. However, the investor can partly mitigate direct transaction costs by trading less, but this increases opportunity costs. Indeed, an investor with a larger AUM internalizes price impact from their trades, and this leads the investor to tilt away from highly predictive but costly-to-trade stocks and signals. Large cost-aware investors opt to forego some predictability in order to hold trading costs

[3]The strategies we develop would be challenging to implement for small investors as they require real-time data on many characteristics across more than a thousand stocks, computation of predictive signals, implementation of ML models, and infrastructure for continually updating and trading these models. Hence, the methods are most relevant for investors large enough to have a staff that can perform these tasks, but, given such capabilities, the implementable investment opportunity set is worse for larger AUM as shown in Figure 1.B.

at bay.

Our paper is related to a large amount of literature. The first applies machine learning methods to enhance return prediction and enhance portfolio performance, including Gu et al. (2020), Freyberger et al. (2020), Chen et al. (2023), Kelly et al. (2019), Gu et al. (2021), Jensen et al. (2023), and Han et al. (2021) in US equity markets; Choi et al. (2021), Leippold et al. (2022), and Cakici and Zaremba (2022) in international equity markets; and Kelly et al. (2022), Bali et al. (2022), and Bali et al. (2021) in bond and derivative markets.

Recent empirical papers point out that trading strategies based on these factors and machine learning methods in asset pricing involve large transaction costs in practice. This literature includes DeMiguel et al. (2020), Li et al. (2020), Chen and Velikov (2021), and Detzel et al. (2021). Motivated by these papers, we develop a flexible portfolio optimization method that lends itself to ML while directly confronting the implementability challenge and explicitly incorporating transaction costs into the ML-based portfolio optimization problem.

Another literature extends the frictionless paradigm of Markowitz (1952) to study portfolio choice in the presence of transaction costs. Constantinides (1986) and Davis and Norman (1990) analyze settings with a single security, where returns are not predictable. Balduzzi and Lynch (1999) and Lynch and Balduzzi (2000) numerically study single-asset trading with predictable returns and transaction costs. Gârleanu and Pedersen (2013) derive an explicit portfolio solution with multiple assets with predictable returns and transaction costs when returns are driven by a factor model. Gârleanu and Pedersen (2016) extend this to more general dynamics in continuous time and Collin-Dufresne et al. (2020) extend the model to include different liquidity regimes. Our contribution is to derive optimal portfolio rules based on stationary dynamics of returns and fully general functional forms for return predictability while incorporating an arbitrarily large set of predictors.

In summary, we provide a theoretical bridge between portfolio optimization and machine learning with powerful empirical results.

1 Model and the Implementable Efficient Frontier

We consider an economy with N securities traded in discrete time indexed by $t = ..., 0, 1, 2, ...$ The return of asset n from time t to $t+1$ is given by $r^f_{t+1} + r_{n,t+1}$, where r^f_{t+1} is risk-free rate and $r_{n,t+1}$ is the asset's excess return. The vector of all assets' excess returns is denoted

r_{t+1}.

An investor observes several characteristics (or signals) for each security, denoted $s_{n,t} \in \mathbb{R}^K$, for example, each asset's valuation ratio, momentum, size, and so on. The characteristics of all assets are collected in the matrix $s_t \in \mathbb{R}^{N\times K}$, and we assume that s_t and r_t are stationary and ergodic. The signals s_t fully characterize the investor's information about returns in the sense that

$$r_{t+1} = \mu(s_t) + \varepsilon_{t+1} \tag{1}$$

where the conditional mean $\mu(s_t) = E_t[r_{t+1}]$ and variance $\Sigma(s_t) = \text{Var}_t[r_{t+1}] = \text{Var}_t[\varepsilon_{t+1}]$ are bounded Borel-measurable functions of s_t with Σ being positive definite.

The investor can be seen as a professional asset manager, such as a hedge fund. The investor has wealth or assets under management (AUM) given by w_t at time t. The asset manager's AUM grows at a stochastic rate, g_t^w, so that $w_{t+1} = w_t(1 + g_{t+1}^w)$, which generally depends on performance and on how clients take money in and out of the fund, as specified in Section 1.2. The investor must choose how much capital, $\pi_{n,t}^{\$}$, to invest in each asset or, equivalently, choose the fraction of the capital invested in each asset, $\pi_{n,t} = \pi_{n,t}^{\$}/w_t$. This portfolio choice implies a dollar profit before transaction costs at time $t+1$ of

$$\text{dollar profit before t-costs}_{t+1} = (r_t^f + r_{t+1})'\pi_t^{\$} + (w_t - 1_N'\pi_t^{\$})r_t^f = w_t(r_t^f + r_{t+1}'\pi_t) \tag{2}$$

where $w_t - 1_N'\pi_t^{\$}$ is the amount of money in the risk-free money market account, and 1_N is a vector of ones. The corresponding return before trading costs, net of the risk-free rate, is

$$r_{t+1}^{\pi,gross} = \frac{\text{dollar profit before t-costs}_{t+1}}{w_t} - r_t^f = r_{t+1}'\pi_t \tag{3}$$

1.1 Trading Costs, Net Returns, and Portfolio Growth

The investor faces transaction costs due to her market impact. Specifically, suppose the investor chooses to trade dollar values given by $\tau_t \in \mathbb{R}^N$ at any time t. This trade leads to a market impact of $\frac{1}{2}\Lambda_t\tau_t$, where $\Lambda_t \in \mathbb{R}^{N\times N}$ is a multivariate version of "Kyle's Lambda."

The resulting transaction cost is the product of the trade size and its market impact:

$$\text{dollar t-costs}_t = \frac{1}{2}\tau_t'\Lambda_t\tau_t\,. \tag{4}$$

Transaction costs are non-negative since Λ is symmetric and positive semi-definite, and transaction costs vary if Λ depends on time and the state s_t of the market.[4]

Given an investor's portfolio choice over time (π_t), the resulting transaction costs are computed by keeping track of the associated trade sizes. At each time, the trade is determined as the difference between the new position and the position "inherited" from the last time period. The inherited position is not just the old position, but the value that the old position has "grown" to in light of the asset returns (or, said differently, the price changes). In particular, the dollar position $\pi^{\$}_{n,t-1}$ bought at time $t-1$ has grown in value to $\pi^{\$}_{n,t-1}(1+r^f_t+r_{n,t})$. Hence, the vector of all dollar trade sizes is

$$\begin{aligned}\tau_t &= \pi^{\$}_t - \text{diag}(1+r^f_t+r_t)\pi^{\$}_{t-1}\\ &= w_t\pi_t - w_{t-1}\,\text{diag}(1+r^f_t+r_t)\pi_{t-1}\\ &= w_t\left(\pi_t - g_t\pi_{t-1}\right)\,,\end{aligned} \tag{5}$$

where, for any vector v, $\text{diag}(v)$ indicates a diagonal matrix with v in the diagonal, and

$$g_t = \text{diag}\left(\frac{1+r^f_t+r_t}{1+g^w_t}\right) \tag{6}$$

is the "growth" of the portfolio weights at time t. Combining equations (2)–(6), we see that the return as a fraction of wealth in excess of the risk-free rate and trading costs is

$$r^{\pi,net}_{t+1} = r^{\pi,gross}_{t+1} - TC^{\pi}_t = r'_{t+1}\pi_t - \frac{w_t}{2}\left(\pi_t - g_t\pi_{t-1}\right)'\Lambda_t\left(\pi_t - g_t\pi_{t-1}\right)\,. \tag{7}$$

where $TC^{\pi}_t = \frac{\text{dollar t-costs}_t}{w_t}$. The portfolio's return naturally depends on the portfolio weights, π, but it also depends on the wealth w_t even though the return is measured in percent of the wealth. This is because trading costs increase by the square of wealth, such that a larger wealth leads to lower portfolio returns after transaction costs. Said differently, a larger

[4]The symmetry of Λ is without loss of generality since, if we start with non-symmetric $\tilde{\Lambda}$, we can define $\Lambda = \frac{1}{2}(\tilde{\Lambda}+\tilde{\Lambda}')$ and note that $\tau'\Lambda\tau = \tau'\tilde{\Lambda}\tau$ for any τ.

investor has a larger market impact (for the same portfolio weights π), thus receiving lower net returns.

1.2 Objective Function

The investor maximizes her long-term average mean-variance utility of portfolio excess returns with relative risk aversion γ_t:

$$\text{utility} = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T} \left[E_t[r_{t+1}^{\pi,net}] - \frac{\gamma_t}{2} \text{Var}_t[r_{t+1}^{\pi,net}] \right] . \tag{8}$$

This objective function corresponds to that of Gârleanu and Pedersen (2013) when the investor is patient (i.e., the discount rate approaches one). The benefit of this long-term-investor objective is that it gives rise to the implementable efficient frontier, as shown below.

We make the following assumptions to keep the problem tractable and stationary. First, the investor has constant risk aversion $\gamma_t = \gamma$, and the risk Σ is constant over time. Second, the investor's wealth (or AUM) grows at an exogenous rate (controlled by how clients take money in and out), so the wealth remains a stationary part of the overall market. Specifically, $w_t\Lambda_t = w\Lambda$, such that the investor faces constant transaction costs relative to her wealth. Under these assumptions, the objective function simplifies as follows

$$\text{utility} = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T} \left(\mu(s_t)'\pi_t - \frac{w}{2} (\pi_t - g_t\pi_{t-1})' \Lambda (\pi_t - g_t\pi_{t-1}) - \frac{\gamma}{2} \pi_t'\Sigma\pi_t \right) \tag{9}$$

$$= E \left[\mu(s_t)'\pi_t - \frac{w}{2} (\pi_t - g_t\pi_{t-1})' \Lambda (\pi_t - g_t\pi_{t-1}) - \frac{\gamma}{2} \pi_t'\Sigma\pi_t \right] , \tag{10}$$

where the investor chooses her portfolio weights $\pi_t = \pi(\vec{s}_t)$ at each time t as a function of all the signals received up until that time, $\vec{s} = (..., s_{t-2}, s_{t-1}, s_t)$, where s_t and g_t are exogenous. Note that while (10) almost looks like a static model, it actually captures dynamic trading as seen from (9): The investor's utility is the average over time (as in (9)) — or, equivalently, her expectation under the stationary distribution (as in (10)) — of the return, net of the cost of trading from π_{t-1} to π_t, and net of disutility of risk.

This setting is ideally suited for a flexible ML implementation for two reasons: First, expected returns are driven by a fully *general function*, μ. Second, the problem is specified in terms of stationary units, namely percentage returns and portfolio weights as fractions

of wealth and a stationary objective function. We assume that μ is uniformly bounded and that investors choose a portfolio policy, $\pi_t = \pi(\vec{s}_t)$, in the admissible set of strategies, Π, that are square integrable, $E[|\pi_t|] < \infty$.

1.3 Empirical Objective: Portfolio Assessments with Trading Costs

The empirical counterpart of our utility function provides a simple and useful way to evaluate real-world investment strategies:

$$\text{utility}_T^{\text{empirical}} = \frac{1}{T}\sum_{t=1}^{T}\left[r_{t+1}^{\pi,gross} - TC_t^{\pi} - \frac{\gamma}{2}(r_{t+1}^{\pi,net} - \bar{r}^{\pi,net})^2\right]. \tag{11}$$

The first part of the sum, $r_t^{\pi,gross} = r'_{t+1}\pi_t$, is simply the average return of the strategy before trading costs. This is the standard metric by which most papers in the literature evaluate trading strategies.

The first and second parts of the sum in (11) give the strategy's net return, $r_{t+1}^{\pi,net} = r_{t+1}^{\pi,gross} - TC_t^{\pi}$. The net return is naturally the gross return minus the trading cost, TC_t^{π}. The trading cost depends on how the portfolio changes from last period, π_{t-1}, to the current period, π_t, that is, $TC_t^{\pi} = \frac{w}{2}(\pi_t - g_t\pi_{t-1})'\Lambda(\pi_t - g_t\pi_{t-1})$.

Finally, when we also include the last part of (11), we get the *return net of trading cost and risk*, $r_{t+1}^{\pi,util} = r_{t+1}^{\pi,gross} - TC_t^{\pi} - \frac{\gamma}{2}(r_{t+1}^{\pi,net} - \bar{r}^{\pi,net})^2$, where the last term captures disutility of risk. This empirical utility simply evaluates the ex-post realized risk, $\frac{1}{T}\sum_{t=1}\frac{\gamma}{2}(r_{t+1}^{\pi,net} - \bar{r}^{\pi,net})^2$, where $\bar{r}^{\pi,net}$ is the average net return.

Our empirical analysis is focused on studying how different investment strategies perform in terms of their empirical utility given by (11). Further, we also compute the corresponding implementable efficient frontier. We note that while standard investment theory implies that the investor should just find the portfolio with the highest Sharpe ratio and later decide on the risk level, this philosophy does not work with trading costs — because the net Sharpe ratio depends on the risk level and the size of the investor, because larger positions have a larger market impact. Instead, we propose that investors focus on their return net of trading cost and risk (11) and their implementable efficient frontier.

1.4 The Implementable Efficient Frontier

The utility function depends on risk and expected returns net of trading costs, which gives rise to the implementable efficient frontier as illustrated in Figure 1 in the introduction. Specifically, we define the implementable efficient frontier as the combination of volatilities and expected net returns, $(\sigma, k(\sigma))_{\sigma \geq 0}$, such that the expected net return is as high as possible for that level of risk:

$$k(\sigma) = \max_{\pi_t \in \Pi} E\left[r_{t+1}^{\pi,net}\right] \quad \text{s.t. } E\left[\pi_t' \Sigma \pi_t\right] = \sigma^2 \tag{12}$$

We are mainly interested in the implementable efficient frontier when taking the maximum among all possible portfolios, but, as seen in Figure 1, we also consider the frontier among subsets, such as standard portfolio sorts.

As an alternative way to compute the implementable efficient frontier, we can derive the optimal portfolio, π^γ, for any level of risk aversion, γ. Based on all these optimal portfolios, we then compute the corresponding combinations of risk and expected net return:

$$(\sqrt{E\left[(\pi_t^\gamma)' \Sigma \pi_t^\gamma\right]}, E[r_{t+1}^{\pi^\gamma,net}])_{\gamma > 0} \tag{13}$$

This generates part of the same implementable efficient frontier, as we show in Proposition 1. The only difference is that, while (12) can generate a downward-sloping curve as seen in Figure 1, (13) only produces a part of the frontier that ends before the downward sloping part, since an investor never wants the dominated portfolios on the downward-sloping part. Proposition 1 characterizes the frontier more broadly:

Proposition 1 (Implementable efficient frontier) *(i) The net Sharpe ratio, $k(\sigma)/\sigma$, is decreasing in σ along the implementable efficient frontier for any level of wealth, $w > 0$, when transaction costs are positive, $\Lambda > 0$; (ii) There exists a critical σ_* such that $k(\sigma)$ is increasing and concave for $\sigma < \sigma_*$; (iii) The part of the frontier $\sigma \in (0, \sigma_*)$ is generated by (13) as $\sqrt{E\left[(\pi_t^\gamma)' \Sigma \pi_t^\gamma\right]}$ decreases in γ and converges to σ_* when $\gamma \to 0$; (iv) If $w_1 < w_2$, then the implementable efficient frontier corresponding to a wealth of w_1 is above that of w_2.*

Interestingly, the implementable efficient frontier has a declining net Sharpe ratio – it is not a straight line with a constant Sharpe ratio as in the textbook frontier without trading costs.

The declining net Sharpe ratio reflects that investors cannot freely leverage their portfolio to the desired risk in the presence of trading costs — because more leveraged positions are larger and incur more significant trading costs. Further, larger investors face higher trading costs, leading to a lower frontier. Propositions 2–7 characterize the implementable efficient frontier at a deeper level via the properties of the underlying portfolios.

2 Solution: Optimal Dynamic Portfolio Choice

We seek to find the optimal portfolio π_t that maximizes average returns net of trading costs and risk (9) or its empirical counterpart (11). The problem is too complex and too high-dimensional to attack by brute force ML of a general function $\pi_t = \pi(\vec{s})$ since $\vec{s} = (..., s_{t-2}, s_{t-1}, s_t)$ is simply of too high dimension. So, we need help from economic theory before we turn to ML.

To solve for the optimal portfolio strategy, we use the "discount factor" m defined in the next lemma. To define this discount factor, we use the notation $\bar{g} = E[g_t]$ for the mean portfolio growth rate as defined in (6), and $G \in \mathbb{R}^{N\times N}$ for the second moments, $G_{ij} = E[g_{it}g_{jt}]$.

Lemma 1 *The fixed point equation*

$$\tilde{m} = \left(w^{-1}\Lambda^{-1/2}\gamma\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}((\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2}) \circ G)\Lambda^{-1/2}\right)^{-1} \tag{14}$$

has a unique, symmetric, positive-definite solution $\tilde{m} \in \mathcal{S}(0,1)$.[5] *For this solution, all eigenvalues of* $\Lambda^{-1/2}\tilde{m}\Lambda^{-1/2}\bar{g}\Lambda$ *are between zero and one. Furthermore,* $m = \Lambda^{-1/2}\tilde{m}\Lambda^{1/2}$ *is such that all eigenvalues of* $m\Lambda^{-1}\bar{g}\Lambda$ *are between zero and one.*

We explain in Appendix B.1 how to calculate the matrix m, but for now, let us treat it as a known constant that depends on the exogenous parameters of the model. Based on this known constant matrix, we can compute the optimal portfolio strategy. We start with a simpler case, namely where expected returns are constant. Even in this case, the solution is non-trivial, as is shown by the following proposition.

[5]The discount factor m is defined by an equation involving the symbol "$\circ$," which is an element-wise matrix product. The element-wise product is also called the Hadamard product, and, for any two matrices A and B, it is computed as the matrix $(A \circ B)_{i,j} = A_{i,j}B_{i,j}$. Further, $\mathcal{S}(0,1)$ is the set of positive-definite matrix-valued functions with eigenvalues between zero and one.

Proposition 2 (Optimal dynamic strategy: constant expected returns) *Let $\tilde{m}$ be the unique solution to (14) in $\mathcal{S}(0,1)$, and let $m = \Lambda^{-1/2}\tilde{m}\Lambda^{1/2}$. When expected returns, $\mu(s_t) = \bar{\mu} \in \mathbb{R}^N$, as well as g_t^w, r_t^f are constant, then the optimal portfolio is given by*

$$\pi_t \;=\; \sum_{\theta=0}^{\infty}\left(\prod_{\tau=1}^{\theta} m\, g_{t-\tau+1}\right)\frac{1}{w}(I - m\Lambda^{-1}\bar{g}\Lambda)^{-1}m\Lambda^{-1}\bar{\mu}\,. \tag{15}$$

Furthermore, π_t is the unique L_2-solution to the stochastic difference equation

$$\pi_t \;=\; m\, g_t\pi_{t-1} + \frac{1}{w}(I - m\Lambda^{-1}\bar{g}\Lambda)^{-1}m\Lambda^{-1}\bar{\mu}\,. \tag{16}$$

To understand the intuition for this proposition, note that the optimal portfolio starts with the old grown position, $g_t\pi_{t-1}$, and then trades toward a fixed portfolio. To understand the direction of the trade, it is useful to write the optimal portfolio as

$$\pi_t \;=\; m\, g_t\pi_{t-1} + (I-m)A = g_t\pi_{t-1} + (I-m)(A - g_t\pi_{t-1})\,, \tag{17}$$

where A is the "aim" portfolio. The aim portfolio has the property that if the investor holds this portfolio, then the investor optimally does not trade, and otherwise, the investor trades in the direction of the aim with trading speed $I - m$. We see from Proposition 2 that the aim portfolio is

$$A \;=\; (I-m)^{-1}(I - m\Lambda^{-1}\bar{g}\Lambda)^{-1}\, c\, \underbrace{\frac{1}{\gamma}\Sigma^{-1}\bar{\mu}}_{Markowitz}\,, \tag{18}$$

where the matrix c is given by

$$c = \frac{\gamma}{w}m\Lambda^{-1}\Sigma \tag{19}$$

So, we see that the aim portfolio is related to the Markowitz portfolio but adjusted to account for transaction costs.[6]

Next, we present a tractable approximation of the optimal portfolio when the expected

[6]We note that, under certain conditions, there is only a small adjustment in the sense that $(I-m)^{-1}(I - m\Lambda^{-1}\bar{g}\Lambda)^{-1}\, c$ is close to I. For example, this happens in the limit when G is close to $\mathbf{1}\mathbf{1}'$. As we show in the Appendix, when $G = \mathbf{1}\mathbf{1}'$, we have $\bar{g} = \text{diag}(\mathbf{1})$, and $c = (I-m)^2$.

returns are not too large. Specifically, we write expected returns as $\mu(s_t) = \epsilon\tilde{\mu}(s_t)$, where $\tilde{\mu}$ is a given function and ϵ is a small number that measures the magnitude of expected returns.

Proposition 3 (Optimal dynamic strategy) *Let $\tilde{m}$ be the unique solution to (14) in $\mathcal{S}(0,1)$ and let $m = \Lambda^{-1/2}\tilde{m}\Lambda^{1/2}$. With expected returns $\mu(s_t) = \epsilon\tilde{\mu}(s_t)$ and $g_t^w = g^w + O(\epsilon)$, $r_t^f = r^f + O(\epsilon)$, the optimal portfolio is*

$$\pi_t \;=\; m\, g_t \pi_{t-1} + (I-m)A_t \;+\; O(\epsilon^2)\,, \tag{20}$$

where the aim portfolio A_t at time t is

$$A_t \;=\; (I-m)^{-1}\sum_{\tau=0}^{\infty}(m\Lambda^{-1}\bar{g}\Lambda)^{\tau}\, cE_t\Big[\underbrace{\frac{1}{\gamma}\Sigma^{-1}\mu(s_{t+\tau})}_{Markowitz_{t+\tau}}\Big]\,. \tag{21}$$

This key theoretical result of the paper shows how to choose the optimal portfolio in two surprisingly simple and intuitive equations. The first equation (20) says that one should always start from the grown position inherited from the last period and then trade toward an aim portfolio.

The second equation (21) shows how the aim portfolio depends on the current and future Markowitz portfolios, thus providing an optimal risk-return tradeoff along the path where these stocks are expected to remain in the portfolio while simultaneously economizing transaction costs.[7] Proposition 3 generalizes the Gârleanu and Pedersen (2013) portfolio optimization principle, "aim in front of the target," when facing trading cost frictions. Unlike Gârleanu and Pedersen (2013), we do not require specific assumptions on return dynamics but, instead, allow a general function $\mu(\cdot)$ that predicts returns. If we make additional assumptions similar to Gârleanu and Pedersen (2013), the aim portfolio (21) can be simplified as shown in Propositon 8 in appendix, but we rely instead on machine learning procedures for the more general formulation.

[7]For the convergence of the series (21), it is important that $m\bar{g}$ has all eigenvalues below one in absolute value. As we show in the Appendix, a stronger claim holds, and $\Lambda^{1/2}\bar{g}^{1/2}m\bar{g}^{1/2}\Lambda^{-1/2}$ is a symmetric, positive semi-definite matrix with all eigenvalues between zero and one.

3 Implementing the Solution with Machine Learning

From a machine learning perspective, Proposition 3 is a powerful result if we assume that s_t is Markovian. To see the power of this result, note that the proposition transfers the ML problem from looking for a general function π of all *current and past* signals to a problem of looking for a function $A(s_t)$ of only the *current* signals. This enormous reduction in dimension vastly simplifies the problem. There are dual ways of using machine learning to implement Proposition 3, which we call "Multiperiod-ML" and "Portfolio-ML," respectively. These approaches are designed to find the dynamic optimal portfolio while being aware of transaction costs in a theoretically consistent manner. We describe each approach in turn.

3.1 Multiperiod-ML: Machine Learning about Expected Returns across Horizons

The first approach to apply Proposition 3 empirically is to compute the aim portfolio using the expected returns across multiple future periods. To understand this approach, recall first that $\mu_t = E_t(r_{t+1})$ is the short-term expected return, so that $E_t[\mu_{t+\tau}] = E_t[E_{t+\tau}[r_{t+\tau+1}]] = E_t[r_{t+\tau+1}]$ is the current expectation about returns τ periods in the future. Using this identity, the aim portfolio can be written as

$$A(s_t) \;=\; (I-m)^{-1}\sum_{\tau=0}^{\infty}(m\Lambda^{-1}\bar{g}\Lambda)^{\tau}c\frac{1}{\gamma}\Sigma^{-1}E_t[r_{t+1+\tau}] \tag{22}$$

The aim portfolio depends on expected returns across all future time horizons.

One approach to apply Proposition 3 empirically is first to use standard ML techniques to predict returns, but do this for a range of forecasting horizons, thus producing proxies for $E_t[r_{t+1+\tau}]$ for all τ. Using these forecasts, the aim portfolio is given by (22). The resulting portfolio can be computed recursively using (20), which we call $\pi^{\text{Multiperiod-ML}}$ as it is based on expected returns over multiple periods.

While many ML methods exist that can be used to forecast returns, we focus on a single method throughout the paper for its unique combination of flexibility and simplicity. Specifically, we use the random features (RF) method of Rahimi and Recht (2007).[8] This method is based on the insight that almost any function can be approximated arbitrarily

[8] Kelly et al. (2022) analyze RF in the context of return prediction and portfolio choice.

well by a linear combination of known auxiliary functions. In other words, we can write $E_t[r_{i,t+1+\tau}] = f(s_{i,t})b_\tau$, where f is a known family of functions of the signals and b_τ is a parameter to be estimated (Section 5.2 details our empirical methodology and Figure C.1 for a simple example). The RF method is powerful in predicting returns and easily adaptable to our second method, which will be discussed next.

3.2 Portfolio-ML: Machine Learning directly about Portfolio Weights

The disadvantage of Multiperiod-ML is that we need a return prediction model for all future return horizons, not just one period ahead. An alternative, and our preferred approach, is to learn *directly* about portfolio weights rather than the two-step procedure of first predicting returns and then constructing portfolios. We thus refer to this approach as "Portfolio-ML."

Based on our objective (9), we let the optimal portfolio π depend on the aim A and then search for the function A that maximizes this objective. This method uses the insights that (i) we can focus on the aim portfolio A, which only depends on current signals, and (ii) the objective should penalize transaction costs.

To see how this works, note that Proposition 3 shows that the optimal portfolio is a weighted average of the inherited position and the current aim portfolio via (20). We can express this result as saying that the current optimal portfolio depends on the current and past aim portfolios and their growth over time:

$$\pi_t \;=\; \sum_{\theta=0}^{\infty} \left(\prod_{\tau=1}^{\theta} m\, g_{t-\tau+1} \right) (I-m) A(s_{t-\theta})\,. \tag{23}$$

So, we can replace π by this expression in the objective function (9), which leaves us the task of finding the best aim portfolio, $A(\cdot)$, based on an economic objective function.

In other words, we need to find a general function $A(s_t)$ that maximizes the expected utility. To do this, we again use the ML insight that any function can be approximated arbitrarily well by a linear combination of known auxiliary functions. Specifically, we write $A_t = f(s_t)\beta$, where f is a set of known functions (random features) of the signals and $\beta \in \mathbb{R}^p$ is an unknown vector of parameters. For example, if portfolio weights were linear in the signals, we could take f to be the identity such that $A_t = s_t\beta$. We take f as a set of random features, just like we did to predict returns (detailed in Section 5.2).

So, we have boiled the portfolio choice problem down to finding the parameter β, and we next show how to do that in closed form. Plugging $A(s_t) = f(s_t)\beta$ into equation (23), we see that the optimal portfolio depends on known elements and the unknown parameter β:

$$\pi_t \;=\; \left[\sum_{\theta=0}^{\infty}\left(\prod_{\tau=1}^{\theta} m\, g_{t-\tau+1}\right)(I-m)f(s_{t-\theta})\right]\beta \equiv \tilde{s}_t\beta\,, \tag{24}$$

where $\tilde{s}_t$ is defined by the last equation. Using this formulation for π_t in the objective (9), we have

$$\begin{aligned}
\text{utility} &= \frac{1}{T}\sum_t \left[r'_{t+1}\pi_t - \frac{\gamma}{2}\pi'_t\Sigma\pi_t - \frac{w}{2}\left(\pi_t - g_t\pi_{t-1}\right)'\Lambda\left(\pi_t - g_t\pi_{t-1}\right)\right] \\
&= \frac{1}{T}\sum_t \left[r'_{t+1}\tilde{s}_t\beta - \frac{\gamma}{2}\beta'\tilde{s}'_t\Sigma\tilde{s}_t\beta - \frac{w}{2}\left(\tilde{s}_t\beta - g_t\tilde{s}_{t-1}\beta\right)'\Lambda\left(\tilde{s}_t\beta - g_t\tilde{s}_{t-1}\beta\right)\right] \\
&= \frac{1}{T}\sum_t \left[\underbrace{r'_{t+1}\tilde{s}_t}_{\equiv\tilde{r}'_{t+1}}\beta - \frac{1}{2}\beta'\underbrace{\left[\gamma\tilde{s}'_t\Sigma\tilde{s}_t + w(\tilde{s}_t - g_t\tilde{s}_{t-1})'\Lambda(\tilde{s}_t - g_t\tilde{s}_{t-1})\right]}_{\equiv\tilde{\Sigma}_t}\beta\right] \\
&\equiv E_T[\tilde{r}'_{t+1}]\beta - \frac{1}{2}\beta' E_T[\tilde{\Sigma}_t]\beta\,.
\end{aligned} \tag{25}$$

So, we can maximize utility by maximizing this quadratic expression in the unknown parameter vector β. To ensure consistency,[9] we add ridge penalty $-\lambda\beta'\beta$, yielding the following solution:

Proposition 4 (Portfolio-ML) *The aim portfolio can be estimated as $A(s_t) = f(s_t)\beta_T$, and the corresponding optimal portfolio, $\pi^{Portfolio\text{-}ML}$, is given by (24), where*

$$\beta_T = (E_T[\tilde{\Sigma}_t] + \lambda I)^{-1} E_T[\tilde{r}_{t+1}] \tag{26}$$

Amazingly, this approach delivers a closed-form solution for the optimal dynamic portfolio in light of transaction costs. To find the optimal portfolio, we compute the two "expectations" on the right-hand side of (26) as their sample counterparts seen in (25). These sample counterparts depend only on data (r_{t+1}, s_t), known parameters, and the ridge parameter λ, which is chosen via ML validation as discussed in Section 5.2. With enough random features and enough time, the estimated portfolio in Proposition 4 asymptotically recovers

[9] Ridge penalization is key for asymptotic consistency of random feature methods. See, e.g., (Beaglehole et al., 2023).

the optimal portfolio, as discussed in more detail in the appendix (Proposition 10).

3.3 Economic Feature Importance

To determine which characteristics are economically important, we consider the value function $V(s)$, the maximum utility for a given set of signals s. We define the importance of any feature n as

$$\iota_n = V(s) - V(s^{-n}) \tag{27}$$

where s^{-n} is the set of signals s, except that we drop feature n at all times. In other words, the importance of feature n is the drop in utility when the investor no longer has access to this information. The following results provide an intuitive characterization of the drivers of feature importance.

Proposition 5 *In the limit when Λ is small, the investor's steady-state optimal utility is $V(s) = v(s) - c(s) + O(\|\Lambda\|^2)$, where*

$$v(s) = \frac{1}{2} E[Markowitz_t' \, \Sigma \, Markowitz_t] \,, \tag{28}$$

is the value function without transaction costs, and c measures the cost of time-variation in the Markowitz portfolio:

$$c(s) = \frac{1}{2} E[(Markowitz_{t+1} - g_{t+1} Markowitz_t)' \, \Lambda \, (Markowitz_{t+1} - g_{t+1} Markowitz_t)] \,. \tag{29}$$

The importance, ι_n, of feature n is

$$\iota_n = \underbrace{v(s) - v(s^{-n})}_{\textit{efficiency loss}} - \underbrace{(c(s) - c(s^{-n}))}_{\textit{cost reduction}} + O(\|\Lambda\|^2) \,. \tag{30}$$

We see that a feature is more important if it is an important contributor to the Markowitz portfolio (the first term in (30)) and if it is a persistent signal such that it reduces the turnover and, hence, the transaction costs (the second term in (30)).

This result provides intuition on the importance of economic features based on an approximation. Empirically, we do not rely on this approximation but, instead, use ML tools

to characterize the economic feature importance (27) as described in Section 6.3.

4 Benchmarks based on Standard Approaches

4.1 Standard Approach: Predicting Returns without T-Costs

The standard approach in the literature is to assume away transaction costs, setting $\Lambda = 0$. In this case, the portfolio problem (9) becomes static in the sense that we can choose the optimal portfolio π_t at time t without regard to what happens at other time periods. Hence, the standard approach is focused on finding methods to predict returns and then using these return predictions to form a portfolio. For example, we can write the standard ML prediction problem as seeking to find a function f of stock characteristics $s_{n,t}$ that minimizes the mean-squared forecast errors for future 1-period (say, 1-month) excess returns $r_{n,t+1}$:

$$\min_{f:\mathbb{R}^K \to \mathbb{R}} \frac{1}{TN} \sum_{n,t} [r_{n,t+1} - f(s_{n,t})]^2 \,. \tag{31}$$

This standard approach generates a function that approximates the conditional mean, $f(s_{i,t}) \cong E[r_{i,t+1}|s_{i,t}]$, when returns are stationary across time and assets, and the number of observations is large. The standard approach to turn such predictions into portfolio weights is to make factor, $\pi^{\text{Factor-ML}}$, by going long a value-weighted average of the top 10% of the assets with the highest predicted returns $f(s_{i,t})$ while shorting the bottom 10% of the assets. Another similar approach is to create a rank-weighted factor, $\pi^{\text{Rank-ML}}$, that invests in each stock in proportion to the rank of its predicted return and scales the resulting weights such that they sum to 1 and -1 on the long and short portfolio, respectively.

These simple factor approaches ignore risk and transaction costs, but a more sophisticated method maximizes (9) while assuming zero transaction costs. Using the vector of expected excess returns $\mu(s_t) = (f(s_{1,t}), ..., f(s_{N,t}))'$, the solution to (9) without transaction costs is

$$\pi_t^{\text{Markowitz-ML}} \;=\; \frac{1}{\gamma} \Sigma^{-1} \mu(s_t) \,. \tag{32}$$

which is an ML-based version of the Markowitz portfolio.

4.2 Static Transaction Cost Optimization

A more sophisticated method is first to estimate the vector of expected excess returns $\mu(s_t)$ via (31) and then account for transaction costs in a second step. While this two-step procedure does not fully account for the dynamic nature of the problem, it serves as an interesting benchmark for our fully dynamic method. To see how this works, consider the problem of choosing an optimal portfolio π_t given the existing portfolio $g_t\pi_{t-1}$:

$$\max_{\pi_t \in \mathbb{R}^N} \left\{ \pi_t' \mu_t - \frac{\gamma}{2} \pi_t' \Sigma \pi_t - \frac{w}{2\phi} (\pi_t - g_t \pi_{t-1})' \Lambda (\pi_t - g_t \pi_{t-1}) \right\}. \tag{33}$$

Here, the transaction costs are divided by a "transaction-cost amortization parameter" ϕ to account for the static nature of the problem in an ad-hoc manner. A naive choice of this parameter is $\phi = 1$, which would mean that the objective (33) compares the returns earned over the next period (a "flow" variable) with the transaction costs (a "stock" variable) paid today. This comparison is problematic if the portfolio is expected to be held for many periods, so having the fudge factor ϕ is a simple way to address this problem. In particular, if the portfolio is expected to be held for $\phi = 3$ periods, we can amortize the trading cost over these three time periods, thus dividing the current transaction cost by three.

The solution to the static objective (33) is:

$$\begin{aligned} \pi_t^{\text{static-ML}} &= (\gamma\Sigma + \frac{w}{\phi}\Lambda)^{-1}(\mu_t + \frac{w}{\phi}\Lambda g_t \pi_{t-1}) \\ &= m^{\text{static}}\, g_t \pi_{t-1} + (I - m^{\text{static}})\, \pi_t^{\text{Markowitz-ML}} \end{aligned} \tag{34}$$

where $m^{\text{static}} = (\gamma\Sigma + \frac{w}{\phi}\Lambda)^{-1}\frac{w}{\phi}\Lambda$. So we see that this strategy is a weighted average of the inherited grown position, $g_t\pi_{t-1}$, and the current Markowitz portfolio. Said differently, this strategy always trades in the direction of the current Markowitz portfolio — so Markowitz is the "aim portfolio" in this static trading cost formulation. This solution is similar to our optimal portfolio with two exceptions. First, the aim portfolio in the fully dynamic model is more forward-looking, distinguishing persistent signals from those with fast alpha decay. Second, the dynamic solution uses the utility optimal discount factor, m, rather than m^{static}.

5 Data and Empirical Methodology

This section describes our data and empirical methodology. In addition, we also make our code available with detailed instructions on how to replicate tables and figures from the paper at https://github.com/theisij/ml-and-the-implementable-efficient-frontier.

5.1 Data and Inputs to the Portfolio Choice

Returns and Investment Universe

We use the dataset from Jensen et al. (2023), a publicly available dataset and replication code of stock returns and characteristics, with the underlying return data sourced from CRSP and accounting data from Compustat.[10]

We restrict our sample to common stocks (`shrcd`: 10, 11, and 12) traded on the New York Stock Exchange (NYSE; `exchcd`: 1). The reason that we do not include stocks traded on AMEX and Nasdaq is that these become available in the CRSP database during our sample, which creates large jumps in the set of available stocks, leading to trading costs. By focusing on NYSE stocks, our empirical analysis is better aligned with our theoretical analysis, which assumes a fixed set of stocks.[11]

To be included in our sample, we require that stocks have non-missing data on market equity, return, dollar volume, and SIC code, and at least 50% of the return predicting signals for the current and the past 11 months. We require this non-missing data over the past 11 months because our Portfolio-ML methodology relies on current and past signals. In our baseline analysis, we further restrict the sample to stocks with a market cap above the median to keep the computational burden manageable, but Section 6.5 shows that our main results also hold when we include all stocks. Finally, stocks are included in the investment universe when they have satisfied the sample restrictions for 12 months and are excluded stocks when they have failed to satisfy the sample restrictions for 12 months. This procedure stabilizes

[10] The code and documentation are available at https://github.com/bkelly-lab/ReplicationCrisis/tree/master/GlobalFactors, and the data is available at https://wrds-www.wharton.upenn.edu/pages/get-data/contributed-data-forms/global-factor-data/.

[11] In a previous version of the paper, we included AMEX and Nasdaq stocks and found very similar results in the baseline analysis with liquid stocks. However, if we restrict attention to the subset of the 20% most illiquid stocks (as in Section 6.5), all methods perform poorly because of the sudden changes in the database with the addition of AMEX stocks in 1962 and Nasdaq stocks in 1972, leading to large artificial turnover. These changes happen during the validation period and, for the illiquid stocks, lead to poor hyper-parameters targeted at mitigating the additions, which are not repeated in the out-of-sample test period.

the sample and reduces trading costs from, for example, stocks that alternate between being above and below the median market cap.

Our sample starts in December 1952 and ends in December 2020, where the first part of the sample is used only for estimation, and our out-of-sample backtests run from 1981 to 2020. When we consider NYSE stocks above the median market cap, the number of stocks in our investment universe reaches a minimum, median, and maximum of, respectively, 184, 646, and 805. When we consider all NYSE stocks, the corresponding numbers are 401, 1270, and 1579.

Signals

To predict returns, covariances, and portfolio weights, we use 115 stock characteristics (or features) studied in Jensen et al. (2023).[12] We standardize each feature in each month by mapping the cross-sectional rank into the [0,1] interval. We set missing values to 0.5 but require at least 57 non-missing features.

Investor Wealth and Optimization Methods

We assume that the investor wealth grows according to the realized market return, $w_t = w_{t-1}(1 + R_{m,t})$, such that the size of the investor is a stable share of the market. This assumption means that the investor withdraws money when the portfolio has outperformed the market and vice versa when the portfolio has underperformed. For interpretability, we label each investor's size by the corresponding wealth level by the end of 2020. In our baseline specification, the investor's wealth by 2020 is \$10 billion.

We assume that the investor optimizes the portfolio each month using the methods we describe below in Sections 5.2–5.3. Some of these portfolio choice methods depend on trading costs and risks, which we estimate each month as described next. While we re-estimate trading costs and risks each month, the investor behaves as if trading costs and risks are constant over time. This assumption simplifies the ML problem, and while it may hurt out-of-sample performance that trading costs and risk do change over time, we find that the methods nevertheless perform well.

[12] Jensen et al. (2023) studies 153 features. However, here, we exclude features with poor coverage early in the sample. Table D.1 shows an overview of the features.

Trading Cost Matrix

Trading cost measured in dollars are given by $\text{TC}_t = \frac{1}{2}\tau_t'\Lambda\tau_t$ for any vector of dollar trades, τ_t. In our empirical analysis, we let the trading cost matrix be diagonal and calibrate it based on the estimates in Frazzini et al. (2018). Specifically, we assume that the market impact, $\frac{1}{2}\Lambda\tau_t$, is 0.1% when trading 1% of the daily dollar volume in a stock. This assumption means that the i^{th} diagonal entry in Λ_t, denoted $\Lambda_{i,t}$, satisfies $0.001 = \frac{1}{2}\Lambda_{i,t}0.01V_{i,t}$, which means that

$$\Lambda_{i,t} = \frac{0.2}{V_{i,t}}, \tag{35}$$

where $V_{i,t}$ is the expected daily dollar volume of stock i at time t. For example, trading \$5 million over a day in a stock with a daily volume of \$500 million moves the price by $\frac{1}{2}\frac{0.2}{\$500\text{m}} \times \$5\text{m} = 0.1\%$, leading to a transaction cost of $\frac{1}{2}\frac{0.2}{\$500\text{m}} \times (\$5\text{m})^2 = \$5000$. We follow Frazzini et al. (2018) and assume that the expected daily volume is equal to the average daily dollar volume over the last six months.

Variance-Covariance Matrix

We need to estimate the variance-covariance matrix, $\Sigma_t = \text{Var}_t(r_{t+1})$, at each time in a way that guarantees it to be positive definite and is broadly consistent with our estimates of expected returns. We use a factor model similar to the MSCI Barra risk model to accomplish these goals. Specifically, security characteristics are used as observable factor loadings, and latent factor returns are estimated via a simple regression (MSCI Barra, 2007).[13] Specifically, each trading day, we estimate a cross-sectional regression of stock returns on stock characteristics

$$r_{i,t+1} = S_{i,t}'\hat{f}_{t+1} + \hat{\epsilon}_{i,t+1}, \tag{36}$$

and the regression coefficients, $\hat{f}_{t+1}$, are the estimated factor returns. Here, the observed characteristics, $S_{i,t}$, consist of 12 dummy variables that indicate the stock's industry and

[13] The procedure of fixing factor loadings and estimating factor returns differs from models such as Fama and French (1993), that fix factor returns and estimate loadings.

13 theme characteristics from Jensen et al. (2023).[14] The theme factors capture the main features that drive return predictions in a simplified way (by reducing more than 100 characteristics to 13 themes and by using a linear factor model rather than ML) and the industry factors capture other forms of covariances. This structure means that the variance-covariance matrix is:

$$\hat{\Sigma}_t = S_t \text{Var}_t(\hat{f}_{t+1}) S_t' + \text{diag}\left(\text{Var}_t(\hat{\varepsilon}_{i,t+1})\right). \tag{37}$$

where the first term is the factor risk and the second term is the idiosyncratic risk.

We estimate $\text{Var}_t(\hat{f}_{t+1})$ as the exponentially-weighted sample covariance matrix of factor returns over the past ten years of daily observations. We weight observations with exponential decays to put more weight on recent observations and, since correlations move slower than variances, we use a half-life of 378 days for correlations and 126 days for variances.[15] Each stock's idiosyncratic variance, $\text{Var}_t(\hat{\varepsilon}_{i,t+1})$, is estimated using an exponentially weighted moving average of squared residuals, $\hat{\epsilon}_{i,s}$, from (36) with a half-life of 126 days. To estimate the idiosyncratic variance we require at least 200 non-missing observations within the last 252 trading days, and otherwise, we use the median idiosyncratic variance of stocks within the same size group.

5.2 Machine Learning Methodology

Machine Learning via Random Fourier Features

We use the machine learning method called random feature (RF) regression from Rahimi and Recht (2007).[16] To understand the intuition behind this method, note that any function

[14] Specifically, each stock's theme characteristic, $S_{i,t}$, is its average rank of the characteristics in the theme, standardized by subtracting the mean and dividing by the standard deviation each month. This standardization implies that the associated factors are long-short and dollar neutral. Regarding the industry factors, we classify SIC codes into 12 industries using the classification from Kenneth French's website. The coefficients on the industry dummies correspond to equal-weighted industry factors.

[15] Specifically, observations j days from t gets a weight of $w_{t-j} = c0.5^{j/\text{half-life}}$ where c is a constant ensuring that the weights sum to one. For comparison, BARRA's USE4 suite of risk models uses a half-life of 84 days for variances in their short-term model, 252 days for variances in their long-term model, and 252 days for correlations in both the short- and long-term models (Manchero et al., 2011, Table 5.1).

[16] See Kelly et al. (2022) for a detailed analysis of the theoretical properties of this RF methodology in the context of return prediction.

$f(s_t)$ can be approximated as

$$f(s_{i,t}) \approx RF(s_{i,t})\beta, \tag{38}$$

where $\beta \in \mathbb{R}^p$ is a vector of parameters and RF consists of random features. The RF method transforms the original features using random weights and a non-linear activation function. There are several ways to generate random features. We use so-called random Fourier features, which essentially approximate a function via its Fourier transformation.[17] While this may sound complicated, it is straightforward to do in practice. We first simply draw some random Normal vectors, $w^j \in \mathbb{R}^{115} \sim iidN(0, \eta^2 I)$ for $j = 1, ..., p/2$. Then, for each j, we create a pair of new features, $\sin(s'_{i,t}w^j)$ and $\cos(s'_{i,t}w^j)$, where the sine and cosine functions can capture non-linearities. We finally collect all these p random features:

$$RF(s_{i,t}) = \frac{1}{\sqrt{p}}\left[\sin(s'_{i,t}w^1), \cos(s'_{i,t}w^1), \ldots, \sin(s'_{i,t}w^{p/2}), \cos(s'_{i,t}w^{p/2})\right]'. \tag{39}$$

Having drawn these random features, the RF method is simply estimating the parameter β from (38) via a ridge regression. The method has two hyper-parameters, namely the number of random features, p, and the standard deviation of the random weights, η, which are chosen via tuning as explained in Section 5.3.

Machine Learning about Expected Returns: Multiperiod-ML

We estimate expected returns using a ridge regression on the RF-transformed features. The resulting model can be viewed as a two-layer neural network with non-optimized weights in the first layer (the random features) and optimized weights in the final layer (the betas). The theoretical solution for Multiperiod-ML in (22) requires expected return over an infinite horizon. Estimating expected returns over an infinite horizon is not practically feasible, so we show how to modify this solution to work with a finite horizon in Appendix B.2. In our empirical analysis, we estimate twelve independent models to predict excess returns in month $t+1, t+2, \ldots,$ and $t+12$.

[17] The approach we use to generate random features is motivated by Sutherland and Schneider (2015), who find that it is preferable to alternative schemes with Gaussian weights.

Machine Learning Directly about the Optimal Portfolio: Portfolio-ML

Our Portfolio-ML learns about the aim portfolio via the relation

$$A_t = f(s_t)\beta = \text{diag}\left(\frac{1}{\sigma_{i,t}}\right) RF(s_t)\beta, \tag{40}$$

where $\beta \in \mathbb{R}^K$ is a parameter, we scale each asset's position by its volatility, $\sigma_{i,t} = \sqrt{\Sigma_{t,ii}}$, and RF consists of random Fourier features as described above. Note the objective for estimation is no longer return prediction but utility maximization, and the solution is given in Proposition 4. The theoretical solution uses the portfolio in (24), which contains an infinite sum of past signals, which we approximate based on the signals from month t to $t-11$.

One practical issue with the aim portfolio in (40) is that the number of stocks varies over time. To avoid that the overall portfolio size does not mechanically increase with the number of available stocks, we standardize each random feature each month as follows. We subtract the random feature's cross-sectional mean and scale it to have a sum of squares equal to one.[18] We also include a positive constant, likewise scaled to have a sum of squares equal to one, to allow the method to control the average level of portfolio weights.

5.3 Portfolio Tuning

The empirical implementation relies on several hyper-parameters as summarized in Table 1. Consider first how we tune our Portfolio-ML method. This method runs a ridge regression on RF-transformed features, so we need to find the ridge parameter λ, the number of random features p, and the standard deviation of random weights η, collected in $h = (\lambda, p, \eta)$.

We tune h as follows. For each h, we compute a "validation backtest" in each year starting in 1971. Specifically, in each year $y \geq 1971$, we compute the optimal β via (26) for that h using monthly data from 1952 to $y-1$. Using this β, we compute the optimal portfolio for each month in year y and repeat this process each year until the end of our sample. This process creates – for each h – a backtest from 1971 onwards. These validation backtests are out-of-sample with respect to the parameter $\hat{\beta}(h)$, but we still need to pick h.

Our "actual backtest" starts in 1981. Each year from 1981 onwards, we pick the hyper-parameter $\hat{h}$ with the highest realized utility in the validation backtest up until now (i.e.,

[18] Specifically, for each month and each random feature, we first demean the feature and then multiply it by $\frac{1}{\sqrt{x'x}}$, where x is the demeaned feature.

Table 1: Hyper-Parameters

Hyper-parameter	Method		
First tuning layer, h	Portfolio-ML	Multiperiod-ML	Static-ML
Ridge penalty, λ	$\{0, e^{-10}, e^{-9.8}, \dots e^{10}\}$	$\{0, e^{-10}, e^{-9.8}, \dots e^{10}\}$	$\{0, e^{-10}, e^{-9.8}, \dots e^{10}\}$
#random features, p	$\{2^6, 2^7, 2^8, 2^9\}$	$\{2^1, 2^2, \dots, 2^9\}$	$\{2^1, 2^2, \dots, 2^9\}$
Std of weights, η	$\{e^{-3}, e^{-2}\}$	$\{e^{-3}, e^{-2}\}$	$\{e^{-3}, e^{-2}\}$
Second tuning layer, h^*		Multiperiod-ML*	Static-ML*
Adjustment to mean, u		$\{0.25, 0.50, 1.00\}$	$\{0.25, 0.50, 1.00\}$
Adjustment to variance, v		$\{1, 2, 3\}$	$\{1, 2, 3\}$
Adjustment to t-cost, k		$\{1, 2, 3\}$	$\{\frac{1}{1}, \frac{1}{3}, \frac{1}{5}\}$

Note: The table shows the hyper-parameter space we use for portfolio tuning. For Portfolio-ML, λ is a ridge penalty, p is the number of random features, and η is the standard deviation of random weights. For Multiperiod-ML* and Static-ML* we add a second tuning layer; u shrinks the expected return vectors as $E_t[r_{t+\tau}]^* = uE_t[r_{t+\tau}]$, v increases stock variances as $\Sigma_t^* = \Sigma_t + v\,\text{diag}(\sigma_t)$, and k controls trading cost as $\Lambda_t^* = k\Lambda_t$.

from 1971 until the previous year). Using this $\hat{h}$ and the corresponding $\hat{\beta}(\hat{h})$, we compute the optimal portfolio over the next year, which is, therefore, truly out-of-sample with respect to both h and β.

For the methods that rely on estimates of expected returns (Multiperiod-ML, Static-ML, Markowitz-ML, Factor-ML, and Rank-ML), we first estimate a model that predicts returns and then compute the optimal portfolio. We predict returns using a similar ML method based on random features using the tuning parameters h shown in Table 1.

We show in the next section that Portfolio-ML outperforms Multiperiod-ML and Static-ML. In fact, the latter methods deliver negative utility to the investor out of sample. This disappointing performance happens even though the ML model to predict returns works reasonably well in terms of how it ranks stocks. The problem is that the resulting portfolios tend to be poorly scaled because out-of-sample returns and risks for optimized portfolios do not match the scale of their ex-ante expected versions. So, this finding already shows the power of the Portfolio-ML method, namely its focus on the economic objective and directly choosing portfolio weights, which immediately leads to an appropriate portfolio scaling with strong performance.

Nevertheless, we want to allow the other methods to compete with the Portfolio-ML method. To improve these alternative methods, we add an additional layer of portfolio tuning to Multiperiod-ML and Static-ML, where we add three additional tuning parameters: u, v, k.

In particular, u shrinks the expected return vector towards zero, v increases the diagonal of the covariance matrix, and k increases the trading cost matrix:

$$\begin{aligned} E_t^*[r_{t+\tau}] &= uE_t[r_{t+\tau}], \\ \Sigma_t^* &= \Sigma_t + v\,\mathrm{diag}(\sigma_t), \\ \Lambda_t^* &= k\Lambda_t. \end{aligned} \tag{41}$$

To estimate two layers of hyper-parameters, we proceed in the following way (which is rather involved, but, again, Portfolio-ML avoids this complexity). In the first layer, we produce a time series of out-of-sample expected returns based on $h = (\lambda, p, \eta)$ and, in the second layer, we produce optimal portfolios based on $h^* = (u, v, k)$. For the first layer, we update the RF models based on h each year using data from 1952 to $y-1$. We estimate the random features models with each set of hyper-parameters from 1952 to $y-11$ and pick the ones that lead to the lowest mean squared error from $y-11$ to $y-1$ (that is, we use the last ten years for validation). After finding the optimal hyper-parameters h, we re-train the model on the full training sample.

In the second layer of portfolio tuning, we update the hyperparameters h^* each year starting in 1981 by choosing the hyperparameters that led to the highest utility since 1971. This two-layer approach is based on some experimentation to make these methods work, which gives these methods an advantage. Again, our main finding is that Portfolio-ML nevertheless performs even better. Figure D.1 in the appendix shows the optimal parameters over time.

6 Empirical Results

This section reports the empirical performance of our novel ML-based investment strategy as well as several other methods for comparison. Our method is implemented via Portfolio-ML (explained in Section 3.2), Multiperiod-ML (Section 3.1), and Multiperiod-ML* with an extra tuning layer (Section 5.3).

As described in Section 4, we compare our strategies to a range of alternative strategies. The simplest alternatives are the market portfolio (buying all stocks in proportion to their market capitalization), the equal-weighted 1/N portfolio of DeMiguel et al. (2009), and

the minimum-variance portfolio. Using the ML-predicted returns over the next month, we compute a standard factor portfolio (Factor-ML), a rank-weighted portfolio that takes positions across all stocks (Rank-ML), and the Markowitz-ML portfolio that also uses the variance-covariance matrix. The most sophisticated alternatives are the Static-ML strategy, which also performs a trading-cost optimization, and Static-ML* with an extra tuning layer (Section 5.3).

6.1 Out-of-Sample Portfolio Performance

Table 2 shows the out-of-sample performance for each method from 1981 to 2020. The performance is computed for an investor with a wealth of $10 billion by the end of 2020 and a relative risk aversion of 10.

Judged by the performance before trading cost, the Markowitz-ML method is the clear winner with an impressive gross Sharpe ratio of 2.06. This finding shows that our methods for predicting risk and return perform well out-of-sample. The gross performance of the trading cost-aware portfolio choice methods (Portfolio-ML, Multiperiod-ML, Static-ML, Multiperiod-ML*, Static-ML*) is substantially lower than that of Markowitz-ML because these methods exploit fewer and less extreme trading opportunities to save on trading costs. Interestingly, several of these methods nevertheless realize a higher gross Sharpe ratio than the standard factor portfolios (Factor-ML, Rank-ML), presumably because they utilize information about both risk and return. Indeed, the minimum variance portfolio, which only uses risk information, also performs well before trading costs.

After accounting for trading costs, the net return of Markowitz-ML, the factor portfolios, and the minimum variance portfolio are highly negative. These methods trade too aggressively and are infeasible for the investor we consider.

In contrast, the trading cost-aware methods still deliver positive net Sharpe ratios, reaching 1.33 for Portfolio-ML, an impressive performance given that we report out-of-sample results and account for the trading cost of a large investor with $10 billion invested.

Turning to our main objective, which is to maximize the realized utility (11), Table 2 shows that the Portfolio-ML approach delivers the highest realized utility. In contrast, the methods in the top panel deliver negative realized utility. This top panel shows our results when all methods are fitted with a single layer of tuning (estimating portfolio weights or expected returns via random feature ML). While Porfolio-ML performs well with a single

Table 2: Out-of-Sample Performance Statistics

Method	R	Vol.	SR_{gross}	TC	R-TC	SR_{net}	Utility	Turnover	Lev.
One tuning layer									
Portfolio-ML	0.15	0.11	1.38	0.006	0.15	1.33	0.086	0.25	3.00
Multiperiod-ML	0.38	0.33	1.15	0.293	0.09	0.26	-0.453	1.70	15.75
Static-ML	0.26	0.28	0.95	0.044	0.22	0.79	-0.165	0.80	12.78
Markowitz-ML	2.78	1.35	2.06	+	-	-	-	79.36	74.61
Factor-ML	0.11	0.13	0.84	2.509	-2.40	-18.09	-2.487	2.79	2.00
Rank-ML	0.08	0.07	1.16	1.266	-1.18	-16.73	-1.210	1.81	2.00
Minimum Variance	0.11	0.11	1.03	0.824	-0.71	-6.41	-0.771	1.35	2.57
1/N	0.10	0.17	0.57	0.004	0.09	0.54	-0.051	0.07	1.00
Market	0.08	0.15	0.52	0.000	0.08	0.51	-0.033	0.01	1.00
Two tuning layers									
Multiperiod-ML*	0.15	0.14	1.10	0.036	0.12	0.83	0.020	0.58	5.69
Static-ML*	0.12	0.10	1.15	0.035	0.08	0.81	0.030	0.68	5.27

Note: The table shows the out-of-sample performance of the various portfolio choice methods, rebalanced monthly from 1981–2020. Here, R is excess return; Vol. is volatility, SR_{gross} is the Sharpe ratio before trading cost; TC is trading cost, R-TC is excess return minus trading cost; SR_{net} is the Sharpe ratio after trading cost; Utility is the realized utility computed as the excess return after trading cost minus one-half times the assumed risk aversion of 10 times the realized portfolio variance; Turnover is the sum of absolute changes in portfolio weights, averaged over time, and Lev. is the portfolio leverage computed as the sum of absolute portfolio weights, averaged over time. All items except turnover and leverage are annualized. Portfolio-ML and Multiperiod-ML are the two dynamic trading cost optimization methods motivated by Proposition 3, Static-ML is the static trading cost optimization method from (34), Markowitz-ML is the optimal portfolio absent trading cost from (32), Factor-ML goes long/short the 10% of stocks with the highest/lowest 1-month expected return, Rank-ML invest in stocks proportional to their ranked 1-month expected return, Minimum Variance is the ex-ante minimum variance portfolio, 1/N invest equally in each stock, and Market invest in stocks proportional to their market equity. For methods with one tuning layers, we search for the optimal hyper-parameters for a ridge regression implemented on RF-transformed features. For Multiperiod-ML* and Static-ML*, we add a second tuning layer to modify expected return, covariance, and trading cost inputs. An entry of "+" or "−" reflects, respectively, an extremely high or low value.

layer of tuning, the other methods deliver negative utility.

It is instructive to consider why Static-ML with a single layer of tuning delivers a negative utility despite its positive net Sharpe ratios. Figure 1.A shows that the indifference curve corresponding to Static-ML goes below the origin, thus yielding a negative utility. This happens because this method realizes a risk that is too high relative to its ex-ante risk estimate. This is seen in Figure 1.A from the fact that the indifference curve crosses the frontier rather than being tangent (we note that the Static-ML frontier is not drawn but has a similar shape as that of Static-ML*). In other words, two factors determine the realized utility (i.e., the return net of trading costs and risk), out of sample: (i) how good the implementable efficient frontier is, and (ii) whether the method places the investor correctly on the frontier based on the investor's risk aversion. While Static-ML produces a frontier

that could deliver positive utility, it places the investor too far to the right on the frontier, thus realizing a negative utility.

To create an even higher hurdle for Portfolio-ML to beat, we also compare its performance to versions of the other methods where we give these other methods an extra "advantage" via a second layer of tuning as described in Section 5.3. This second layer of tuning is designed to improve the scaling of the portfolio, thus helping these methods to place the investor more correctly on the implementable efficient frontier.

Table 2 shows that the two-layer versions actually deliver positive utility. This result highlights that the second tuning layer is crucial for these methods based on ML about expected returns. Nevertheless, our Portfolio-ML continues to outperform these methods. This outperformance of Portfolio-ML relative to the two-layer methods shows a benefit of learning directly about portfolio weights, namely that the ML algorithm immediately searches for a well-scaled portfolio that delivers high utility – so no additional tuning layer is needed.

As seen from the notation in Table 2, we add a superscript "$*$" to the implementations with two tuning layers. In the remainder of this section, we compare Portfolio-ML and the two-layer alternatives, Multiperiod-ML* and Static-ML*, studying their performance over time and the statistical significance of their performance differences.

Figure 2 shows that the performance over time, measured in terms of gross returns, net returns, and utility (return net of trading costs and risk). The figure shows that Portfolio-ML outperforms Static-ML relatively consistently in all three respects. Figure 2 also shows interesting time-series patterns, such as a relatively lower performance during the dot-com bubble in 2000, the global financial crisis in 2008, and the COVID-19 crash in 2020.

One of the reasons behind the outperformance of Portfolio-ML is that this method keeps trading costs lower. This lower trading cost is achieved via a lower monthly turnover of 25% relative to 58% and 68% for Multiperiod-ML* and Static-ML*, respectively, as seen from Table 2.

Another way Portfolio-ML achieves lower trading costs is by focusing on more liquid stocks. This point is illustrated in Figure 3, which shows position sizes by stock liquidity groups for the market portfolio, Markowitz-ML, and Portfolio-ML. Naturally, the market portfolio takes larger positions in more liquid stocks. In contrast, Markowitz-ML is agnostic to liquidity and actually tends to take slightly larger positions in the least liquid stocks because of the more predictable return among these. Portfolio-ML displays a compromise

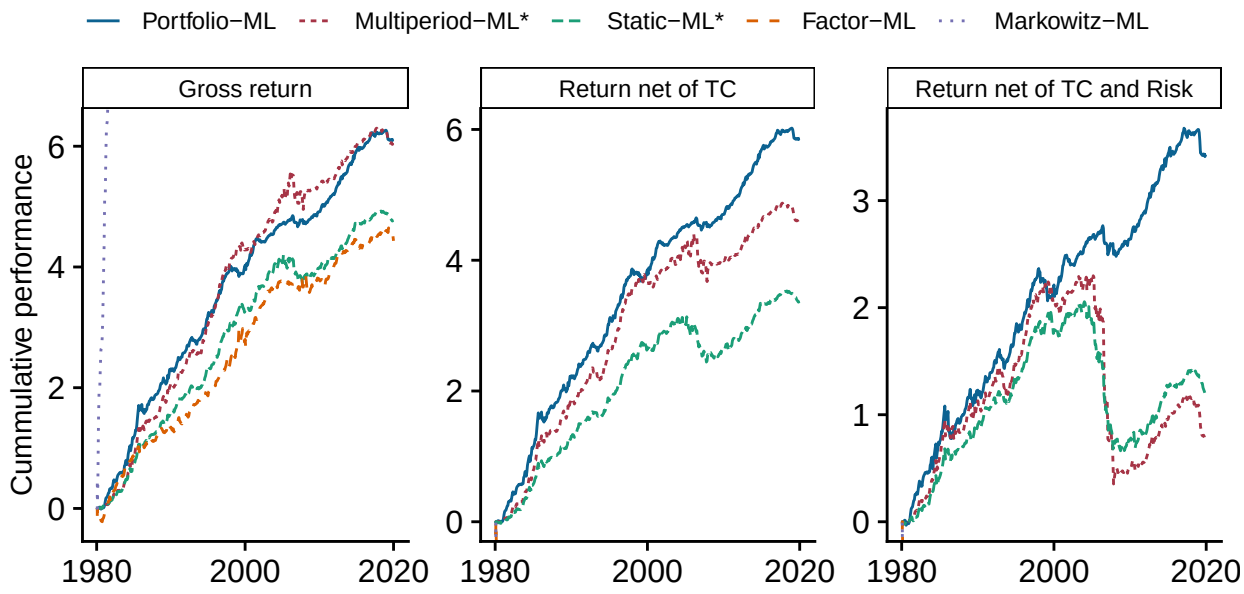

Figure 2: Performance over Time

Note: The left panel shows the cumulative sum of returns before trading cost, $r_{t+1}^{\pi,gross}$, for each portfolio method. The middle panel shows the cumulative sum of returns net of trading cost, $r_{t+1}^{\pi,net}$. The right panel shows the cumulative return net of trading cost (TC) and net of disutility from risk, computed as $r_{t+1}^{\pi,util} = r_{t+1}^{\pi,gross} - TC_t^{\pi} - \frac{\gamma}{2}(r_{t+1}^{\pi,net} - \bar{r}_{t+1}^{\pi,net})^2$, corresponding to the realized utility. We assume that the investors has a relative risk aversion of 10 and invested wealth of \$10 billion by the end of 2020.

between focusing on liquid stocks to reduce trading costs while simultaneously trying to diversify across stocks to exploit various sources of return predictability.

To visualize an example of some specific portfolio weights over time, Figure 4 depicts how the portfolio weights for Johnson and Johnson and Xerox stocks evolve for each method. Portfolio-ML adjusts its positions more slowly than the other methods, especially for the less liquid stock (Xerox).

Table 3 reports the statistical significance of the relative performance differences across portfolio choice methods. Specifically, the table reports the Bayesian probability that each method outperforms any of the other methods. To compute these pairwise probabilities of one method outperforming another, we first compute the utility flow (return net of trading costs and risk) of each method π at time $t+1$ as $r_{t+1}^{\pi,util} = r_{t+1}^{\pi,gross} - TC_t^{\pi} - \frac{\gamma}{2}(r_{t+1}^{\pi,net} - \bar{r}_{t+1}^{\pi,net})^2$, where $\bar{r}_{t+1}^{\pi,net}$ is the average return after trading costs over the full sample and the relative risk aversion is $\gamma = 10$ as before. We then compute the utility difference between any two methods, say π and $\tilde{\pi}$, as $d_{\pi,\tilde{\pi},t+1} = r_{t+1}^{\pi,util} - r_{t+1}^{\tilde{\pi},util}$. If the difference is normally distributed with a non-informative prior mean and a known prior variance, the posterior of the true

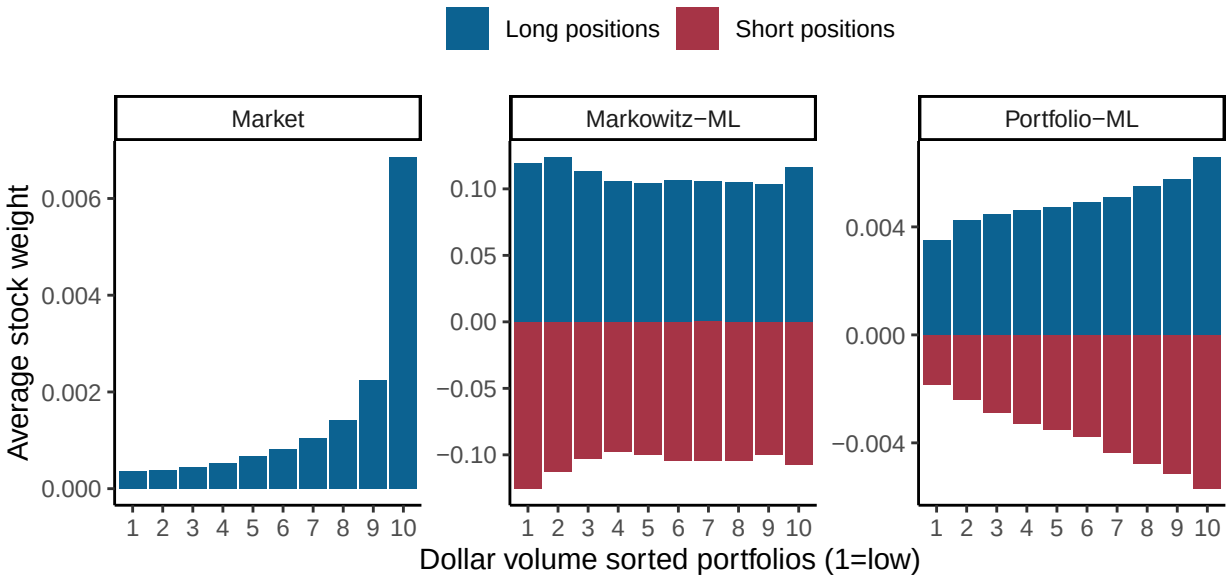

Figure 3: Position by stock liquidity

Note: The figure shows the position taken in stocks as a function of their liquidity for the market portfolio, Markowitz-ML, and Portfolio-ML. For each method, we compute the average portfolio weight of stocks within each of the ten liquidity groups (based on our estimated trading cost) separately for stocks held long and short. The figure shows this number averaged over time, from 1981 to 2020.

utility difference is then normally distributed with mean $\bar{d}_{\pi,\tilde{\pi}} = \frac{1}{T}\sum_{t=1}^{T} d_{\pi,\tilde{\pi},t}$ and variance $\frac{1}{T-1}\sum_{t=1}^{T}(d_{\pi,\tilde{\pi},t}-\bar{d}_{\pi,\tilde{\pi}})^2$. Based on these calculations, Table 3 reports the posterior probability that $d_{\pi,\tilde{\pi}} > 0$, that is, the posterior probability that the first portfolio choice method, π, delivers a higher average utility than the second method, $\tilde{\pi}$.

Table 3 shows that the probability that Portfolio-ML delivers a higher expected utility than Multiperiod-ML*, Static-ML*, Factor-ML, and Markowitz-ML are, respectively, 99.9%, 99.9%, 100%, and 100%, suggesting that the superiority of Portfolio-ML is not just random noise.

Alternatively, we can think of the probabilities in Table 3 as being approximately the p-value of a one-sided test that the realized utility of i is greater than j. Hence, we see that we can reject that Portfolio-ML delivers a lower realized utility than the other methods at conventional levels of significance.

Lastly, Table 4 reports the return correlations of the various portfolio choice methods. We see that all methods are positively correlated, but the magnitudes are modest. In addition to showing the relative connection across these methods of portfolio choice, these findings may also be informative about asset pricing more generally. Indeed, the Markowitz-ML portfolio

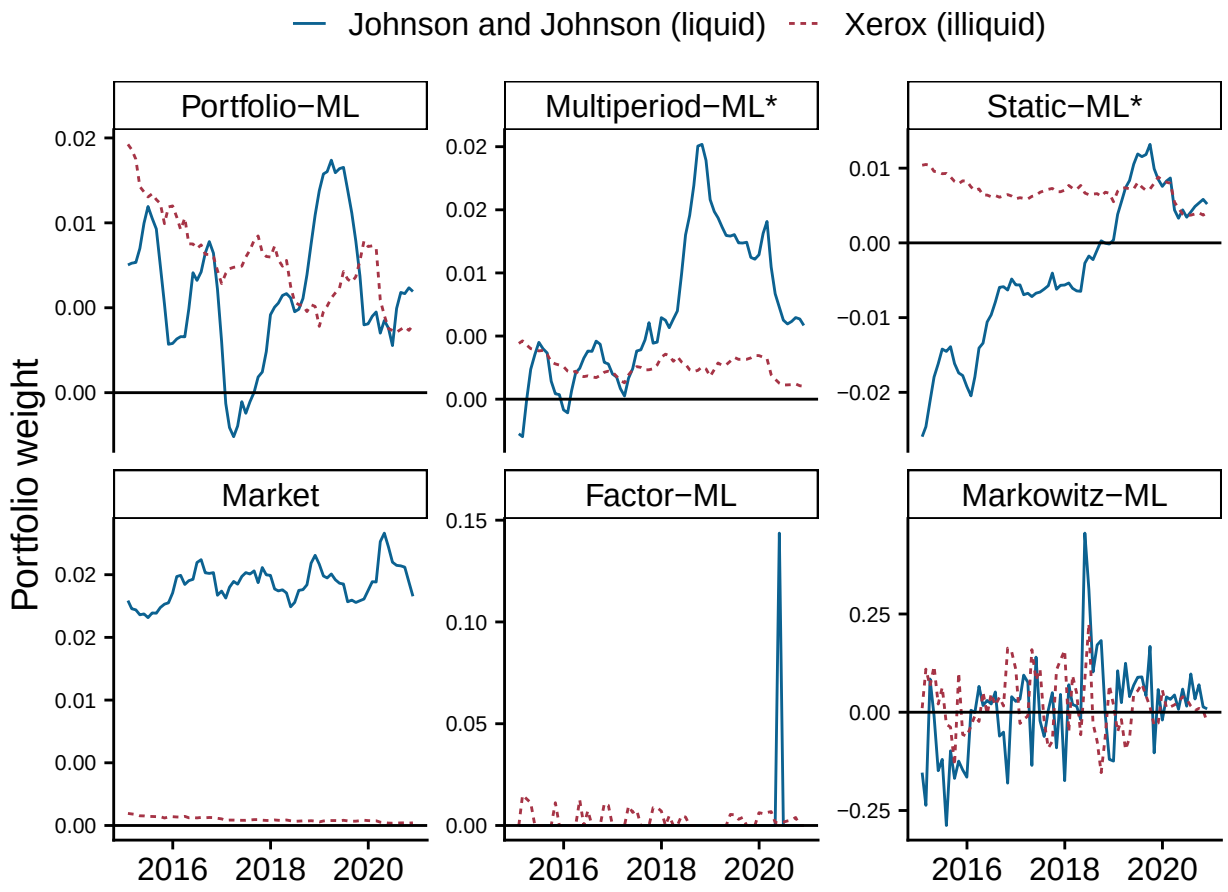

Figure 4: Portfolio Weights: Johnson and Johnson vs. Xerox

Note: The figure shows the portfolio weights of Johnson and Johnson and Xerox for six portfolio choice methods, 2015–2020. Johnson and Johnson was chosen as an example of a relatively liquid stock, and Xerox was chosen as a relatively illiquid stock over this time period. By the end of 2020, the average daily dollar volume over the past six months was $965M for Johnson and Johnson and $55M for Xerox.

return can be viewed as an estimate of the minimum-variance stochastic discount factor (SDF) in a frictionless market as shown by Hansen and Jagannathan (1991). Since risk adjustments depend on covariance with the SDF, a natural question is how closely the SDF aligns with the corresponding measure designed for a market with frictions. The correlation between Portfolio-ML and Markowitz-ML is only 0.26, indicating that the marginal utility of an investor with $10 billion using Portfolio-ML could be very different from risk adjustments in a frictionless market.

In summary, this section shows that Portfolio-ML outperforms the other methods, delivering a high net Sharpe ratio and high utility. While this strong performance suggests that Portfolio-ML works well, a few words of warning are in order. First, while the net

Table 3: Relative Probability of Outperformance

	Portfolio-ML	Multiperiod-ML*	Static-ML*	Factor-ML	Markowitz-ML
Portfolio-ML		99.9%	99.9%	100.0%	100.0%
Multiperiod-ML*	0.1%		27.1%	100.0%	100.0%
Static-ML*	0.1%	72.9%		100.0%	100.0%
Factor-ML	0.0%	0.0%	0.0%		100.0%
Markowitz-ML	0.0%	0.0%	0.0%	0.0%	

Note: The table shows the probability of the row method having a higher average utility than the column method. The probability is computed via an uninformative prior, assuming that the difference in utility is normally distributed. The utility flow of any method π at time $t+1$ is $r^{\pi,util}_{t+1} = r^{\pi,gross}_{t+1} - TC^{\pi}_{t} - \frac{\gamma}{2}(r^{\pi,net}_{t+1} - \bar{r}^{\pi,net}_{t+1})^2$. One can also think of each number as the p-value in the test of whether the average utility of the portfolio choice method in the row is greater than the average utility of the method in the column.

performance is extremely good in our simulation, real-world investors seeking to achieve this performance must often pay fees to an asset manager (e.g., a hedge fund) running such strategies and face other real-world complications, potentially reducing performance. Second, investors might not have been able to realize this performance in real-time due to more limited computing power and a less developed ML methodology in the early sample. In any case, this warning applies to any simulation, and the statistically significant outperformance of Portfolio-ML relative to other methods is an encouraging apples-to-apples test.

6.2 Evidence on the Implementable Efficient Frontier

Textbooks and real-world investors often depict their investment opportunities in terms of the achievable combinations of risk and expected return. This illustration highlights that investors seek a portfolio on the efficient frontier with the highest expected return for any level of risk. The textbook version of the efficient frontier — in sample, without trading costs — is a straight-line tangent to the hyperbola of risky investments. However, we propose that investors should focus on what we call the implementable efficient frontier — based on out-

Table 4: Portfolio Correlations

	Portfolio-ML	Multiperiod-ML*	Static-ML*	Factor-ML	Markowitz-ML
Portfolio-ML	1.00				
Multiperiod-ML*	0.61	1.00			
Static-ML*	0.52	0.80	1.00		
Factor-ML	0.26	0.35	0.40	1.00	
Markowitz-ML	0.26	0.48	0.58	0.40	1.00

Note: The table shows the time-series correlation of the returns before trading costs for the various portfolio choice methods, 1981-2020.

of-sample performance net of trading costs.

Figure 1 illustrates our estimated implementable efficient frontier. To understand how we have generated this plot, we start by describing the two benchmarks for a world without trading costs. The hyperbola is a mean-variance frontier of risky assets inspired by Markowitz (1952, 1959). We generate the points on the frontier by minimizing variance for a given mean and requiring that portfolio weights sum to 1:

$$\begin{aligned} \min_{\pi_t \in \mathbb{R}^N} \quad & \pi_t' \Sigma_t \pi_t, \\ \text{s.t.} \quad & \pi_t' \mu_t = k, \\ & \pi_t' 1_N = 1, \end{aligned}$$

where k is the required mean return, and 1_N is a vector of ones. The solution is given by

$$\pi_t = \frac{c_t k - b_t}{d_t} \Sigma_t^{-1} \mu_t + \frac{(a_t - b_t k)}{d_t} \Sigma_t^{-1} 1_N, \tag{42}$$

where $a_t = \mu_t' \Sigma_t^{-1} \mu_t$, $b_t = 1_N' \Sigma_t^{-1} \mu_t$, $c_t = 1_N' \Sigma_t^{-1} 1_N$, and $d_t = a_t c_t - b_t^2$ are constants. Implementing this solution for a range of k's generates the hyperbola. A standard presentation of the frontier uses one cross-section of stocks (i.e., one μ_t and one Σ_t) and presents the ex-ante expected frontier. In contrast, we show the realized frontier out-of-sample. Specifically, for each k, we update portfolio weights each month using (42). We then record the realized return and volatility before trading cost over the sample, 1981-2020. In contrast to the standard textbook presentation, the efficient frontier of risky assets in Figure 1 also accounts for out-of-sample performance decay. As such, it gives a more realistic picture of what investors could achieve absent trading costs.

The second benchmark for the case without trading cost is the Markowitz-ML portfolio. In a standard presentation, the line from this portfolio would be tangent to the hyperbola. However, because our analysis is out-of-sample, this outcome is not ensured. Specifically, the portfolio with the highest estimated Sharpe ratio ex-ante is not necessarily the one with the highest realized Sharpe ratio ex-post. In our analysis shown in Figure 1.A, Markowitz-ML is approximately tangent, suggesting that it is a good benchmark for the best possible portfolio without trading cost.

Next, we shift the attention from the frictionless benchmarks to our main focus, namely

the implementable efficient frontier. We illustrate the performance of Markowitz-ML and Factor-MLs with trading cost by scaling each portfolio to ex-post volatilities ranging from 0 to 10% in increments of 1%. We see their returns after trading costs are negative, except at very low volatilities. Hence, implementable efficient frontiers of these standard methods show that an investor maximizes utility by putting almost all wealth into the risk-free asset, thus choosing to hardly trade on these standard methods.

Turning to the trading-cost-aware methods, Portfolio-ML and Static-ML*, we see that their performance is much better. Instead of varying the ex-post volatilities, we implement the methods under five different relative risk aversions, $\gamma \in \{1, 5, 10, 20, 100\}$, and interpolate between their realized performance to plot an efficient frontier via equation (13). Comparing the two implied frontiers, we see that Portfolio-ML leads to a higher achievable return for the same volatility. More generally, the figure shows that it is feasible for a large investor to generate an attractive, implementable, efficient frontier, even net of trading costs.

Panel B of Figure 1 shows how the implementable efficient frontier varies by investor size. Specifically, we implement the Portfolio-ML method for the same relative risk aversions as above, but now we also vary the investor wealth, $w_{2020} \in \{0, 10^9, 10^{10}, 10^{11}\}$. Such a plot is not interesting without trading costs since the efficient frontier is the same regardless of investor size. With trading costs, this is no longer the case. Price impact is increasing in trade size, so a larger investor must trade more slowly and focus more on liquid stocks. Naturally, these effects imply that larger investors have a worse risk-return tradeoff. The results in Figure 1.B quantifies how much worse. The figure shows the substantial cost of being a large investor. For example, an investor with \$10B and a relative risk aversion of 10 gets a net excess return of 14.6% at 11% volatility. If the investor had \$1B instead, the same volatility would provide a net excess return of 16.3%. In summary, once we introduce trading costs, we no longer have a unique, efficient frontier. Instead, the implementable efficient frontier depends on the investor's size and portfolio choice method.

6.3 Economic Feature Importance

Following the theoretical discussion in Section 3.3, we define economic feature importance as the drop in utility when excluding a feature (i.e., signal) from the information set of the investor. To implement this idea, we use the concept of "permutation feature importance," introduced by Breiman (2001), a standard method for assessing the feature importance of

any machine learning model (Molnar, 2022). The basic idea is to permute features randomly and assess the decline in the value function. As such, rather than actually excluding a feature from the information set, the method instead destroys any predictive relationship between the feature and the outcome variable, which has the benefit that we do not need to re-estimate the model.

Given that the objective function is a utility function, this method yields a measure of economic feature importance. To compute the feature importance of any signal j for any investment policy, say Portfolio-ML, we first compute the baseline realized utility, utility$[\pi^{\text{Portfolio-ML}}(s)]$, of the portfolio, $\pi^{\text{Portfolio-ML}}$, computed based on the original features, s. Next, for the feature in question, j, we randomly permute its associated values at each given time while keeping all other features at their actual values. We then implement the portfolio method using the same parameters as in the original specification, but now with the permuted features, $s_{\text{perm},j}$, as inputs. Finally, we compute the economic feature importance as the resulting drop in utility,

$$\text{FI}_j^{\text{Portfolio-ML}} = \text{utility}[\pi^{\text{Portfolio-ML}}(s)] - \text{utility}[\pi^{\text{Portfolio-ML}}(s_{\text{perm},j})] \tag{43}$$

In other words, a feature j is economically important for an investor using Portfolio-ML if destroying its informational content leads to a large drop in realized utility.

A potential issue with permuting each feature separately is that substitution effects can distort the inference. For example, we include many different value features, so the effect of permuting a specific feature, such as book-to-market, is muted because the method can rely on other value features, such as assets-to-market or earnings-to-price. To handle substitution effects, we permute *all* features within a specific theme and record feature importance at the theme level. We use the 13 themes from Jensen et al. (2023), shown in Table D.1 in the appendix.

Figure 5 shows each theme's feature importance for two investment policies. The left panel shows feature importance after trading costs for a large investor with a wealth of \$10b by 2020 for Portfolio-ML. The right panel shows feature importance without trading costs for Markowitz-ML, where we ignore trading costs because this method does not work after trading costs, making it meaningless to discuss net-of-cost feature importance. As such, the right panel serves as the benchmark of a frictionless market.

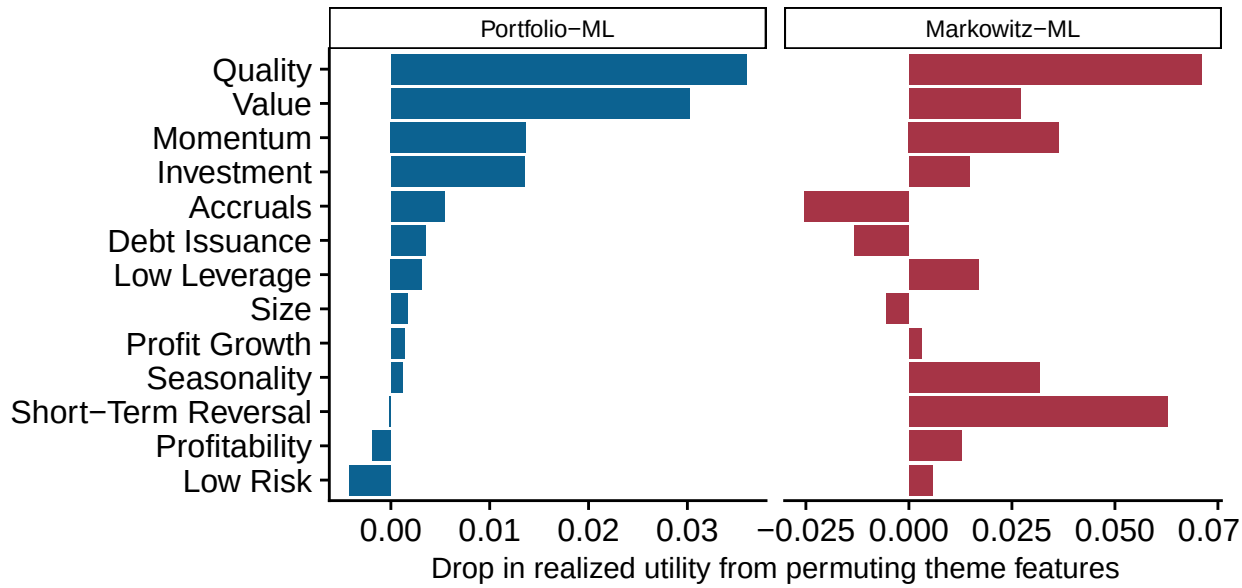

Figure 5: Economic Feature Importance

Note: The figure shows a utility-based feature importance measure for Portfolio-ML and Markowitz-ML. We randomly shuffle the associated features for each theme while keeping all other features at their actual value. We then implement each method based on this counterfactual data and measure feature importance, such as the difference in realized utility relative to the implementation that uses the actual data. For Markowitz-ML, we assume that the investor can trade without incurring the trading cost.

Looking at the frictionless benchmark in the right panel of Figure 5, we see that the important feature themes before trading costs are quality and short-term reversal. Turning to the net-of-cost importance of the feature in the left panel, we see that quality remains important for a large investor. In contrast, short-term reversal is far less important due to the high turnover of many factors within this theme (see figure D.2). This finding is consistent with the theoretical results of Section 3.3, namely that high-frequency features are less important in the presence of trading costs.

For example, the short-term reversal theme includes the one-month reversal factor, which has a monthly autocorrelation of -0.04. This autocorrelation is not just low but actually negative, giving rise to large portfolio turnover.[19] For a large investor, the predictive ability of short-term reversal is not enough to overcome the cost of trading it. In comparison, book-to-market has a monthly autocorrelation of 0.94, indicating that it is a highly persistent feature, thus economizing on trading costs.

For Portfolio-ML, quality emerges as the most important theme. The median monthly

[19]The autocorrelation of a feature is computed as the average autocorrelation across all stocks with at least five years of monthly observations.

autocorrelation of quality features is 0.93, so the result is again consistent with the theoretical findings. Perhaps surprisingly, momentum, generally considered a "fast" signal, is also one of the more important themes for a large investor. However, several momentum features actually do exhibit meaningful persistence; for example, 12-month return momentum has a monthly autocorrelation of 0.87. Furthermore, momentum and value are negatively correlated, which leads to less trading because the two signals offset each other. Finally, investment is the fourth most important, after which there is a substantial drop to the next-most important themes.

The implementable efficient frontier can also be used to illustrate economic importance in an intuitive way. For a specific theme, say quality, we compute the out-of-sample frontier for an investor who uses counterfactual data in which the quality signals have been randomly shuffled. We can then compare this frontier with the original frontier, which does not shuffle the data. Figure 6 illustrates economic feature importance in this way.

Panel A the net-of-trading-cost frontier generated by Portfolio-ML, both the original frontier and the frontiers with shuffled quality, value, and short-term reversal, respectively. We see that, for a large investor, quality and value signals are important for the implementable frontier. In contrast, destroying the informational content of short-term reversal signals barely changes the achievable frontier because the trading-cost-aware Portfolio-ML method hardly uses this signal anyway.

Panel B shows the impact on the before-trading-cost frontier generated by Markowitz-ML. In this case, all three themes are important. Interestingly, while short-term reversal has a minor effect on the implementable frontier with trading costs in Panel A, it greatly impacts the frontier without trading costs in Panel B.

6.4 Short-Selling Costs

So far, we have focused on the frictions generated by trading costs, but real-world investors also face short-selling fees. To assess the impact of short-selling fees, we use data from Markit covering 2003–2020. We focus on the "indicative fee," which is an estimate of the shorting fee for a specific stock on a specific day, and extend this indicative fee to all stock months to avoid missing data.[20]

[20]The indicative fee is only available 2017-2020, but throughout 2003-2020, most stocks in our investment universe have a "daily cost of borrow score" (DCBS), which is a score from 1 to 10 (low to high shorting

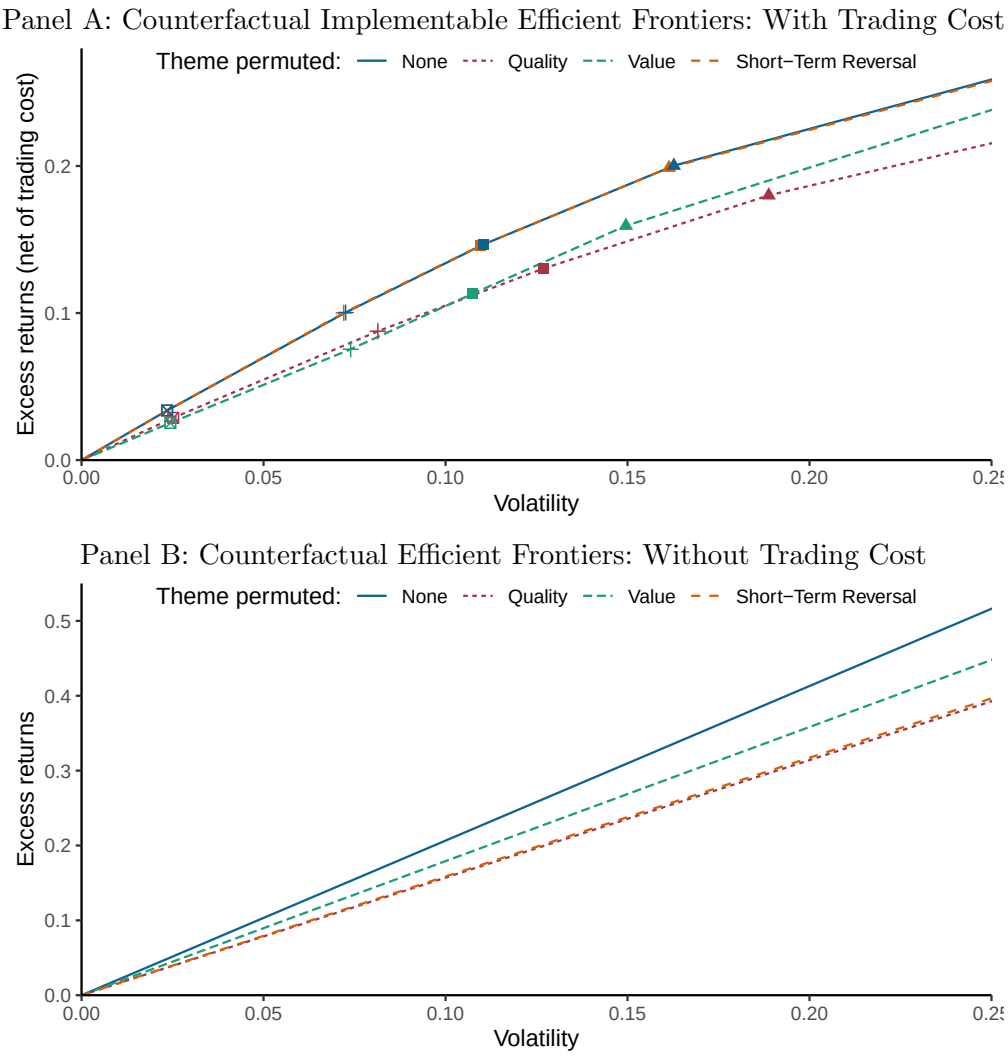

Figure 6: Feature Importance: Counterfactual Implementable Efficient Frontiers

Note: Panel A shows the implementable efficient frontier with trading costs for an investor with a wealth of \$10B by 2020. We implement the Portfolio-ML method using a counterfactual data set, where we permute all feature values related to either quality, value, or short-term reversal. The solid blue line shows the frontier using the actual data. Panel B shows the same analysis without trading cost, now using the Markowitz-ML method. In both panels, the relative risk aversions are 1 (circle), 5 (triangle), 10 (square), 20 (plus), and 100 (boxed cross), and the sample period is 1981-2020.

Figure 7 shows the impact of shorting fee for the realized utility from Portfolio-ML, Multiperiod-ML*, and Static-ML*. The first bar shows the realized utility without shorting fees from Table 2. The second bar shows the realized utility after subtracting short-selling fees. The annualized cost is 0.5% for Portfolio-ML, 1.2% for Multiperiod-ML*, and 1.1% for

fees). We extend the data as follows: (i) if the indicative fee is available, we use it; (ii) if the indicative fee is missing and DCBS is non-missing, we use the median indicative fee of stocks with the same DCBS; (iii) If both the indicative fee and DCBS are missing, we will use the median indicative fee among all stocks.

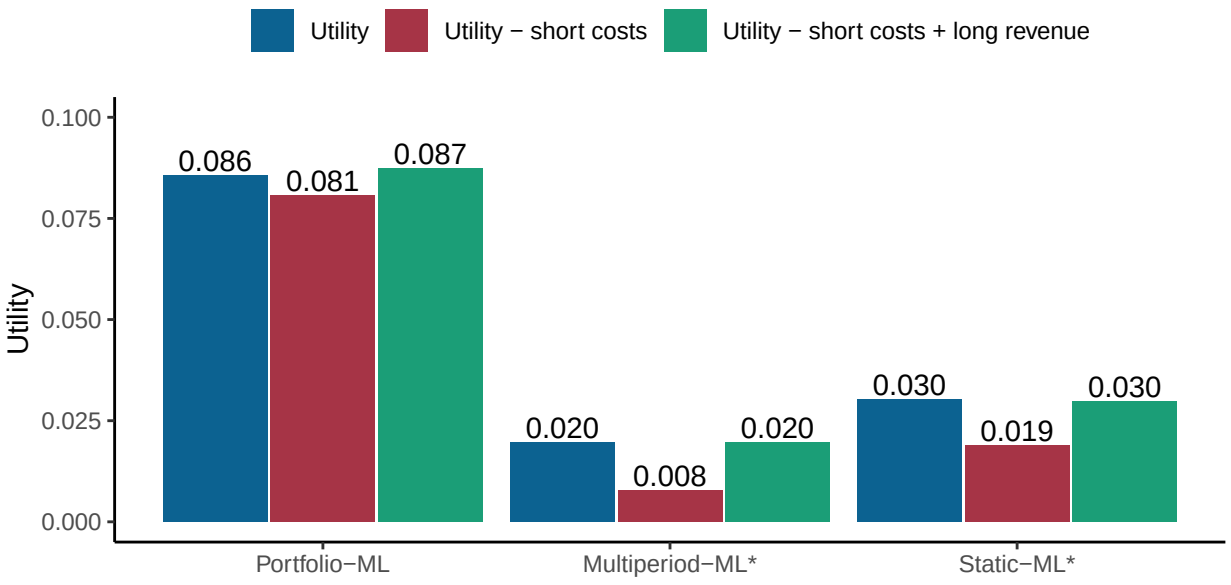

Figure 7: Short-selling costs

Note: The first bar shows the annualized utility (out-of-sample return net of trading costs and dis-utility from risk) without considering shorting costs, the second bar shows the utility minus the average annualized cost of shorting stocks (i.e., stocks that have a negative portfolio weight), and the third bar adds back the average annualized revenue from lending out stocks that are held long (i.e., have a positive portfolio weight). The short-selling fees are based on indicative fees from Markit, and we assume that the investor pays 100% of the cost for the stocks they sell short and receives 70% of the lending fees on stocks that are long.

Static-ML*, leading to corresponding meaningful, but modest, drops in utility.

To understand these results, note that shorting costs are low for the vast majority of stocks in our investment universe, which consists are large and liquid stocks. For example, by the end of our sample in December 2020, the median shorting fee is 0.30% per year and the 99th percentile is 1.1%. Therefore, shorting fees have a modest impact on our results.

Furthermore, net securities lending costs are even smaller for investors who choose to lend out the shares that they hold long. When lending out their shares, investors receive part of the fees that short-sellers pay to borrow the shares. Following Muravyev et al. (2022), we assume that investors keep 70% of the securities lending fee, while the rest go to the brokers that facilitate the securities lending transaction.

The realized utility after adding back the proceeds from securities lending is shown in the third bar in Figure 7. We see that this realized utility is almost identical to the performance before considering shorting cost. This result might be surprising given that the investor only receives 70% of the shorting fee on the long side. However, these portfolios hold more stocks long than short, which ends up counteracting this effect. All in all, this analysis suggests

that the methods that we consider remain profitable after shorting costs.

6.5 Performance across the Size Distribution

In the baseline analysis, we only consider relatively large stocks, namely those with a market cap above the NYSE median. This choice is made mainly to ease the computational burden, but here we show that the main results also hold when we consider all stocks and stocks in different size groups.

Using the sample of all NYSE stocks, the first panel in Figure 8 shows each investment strategy's realized utility (11), i.e., the average annual return net of trading costs of disutility from risk. We see that, as in the baseline sample, Portfolio-ML is the best-performing method, followed by Multiperiod-ML* and Static-ML*, and all other methods realize negative utility. The biggest difference relative to the baseline results in Table 2 is that Multiperiod-ML* now clearly outperforms Static-ML*.

Figure 8 also shows the results when we partition stocks into five groups based on their market capitalization. The relative ranking is consistent in each of the four largest size groups, with Portfolio-ML performing the best, followed by Multiperiod-ML* and Static-ML*.

For the smallest stocks, Multiperiod-ML* and Static-ML* are the best-performing methods, followed by Portfolio-ML. This finding is surprising since the benefit of dynamic strategies over static ones might be especially valuable among illiquid stocks. However, the smallest stocks also suffer from more changes in the investment universe (breaking the dynamics as stocks disappear), and lower data quality may have a larger impact on more sophisticated methods. That said, it is comforting that the dynamic methods continue to deliver positive utility.

To further analyze the performance in a less liquid investment universe, we simulate an economy where we know the true data-generating process in Appendix E. In this simulated economy, stocks do not disappear, the data quality is high, and the data-generating process is stable. Consistent with our theory, we indeed find a large benefit of Portfolio-ML and Multiperiod-ML* relative to Static-ML*, and this benefit is larger for more illiquid stocks.

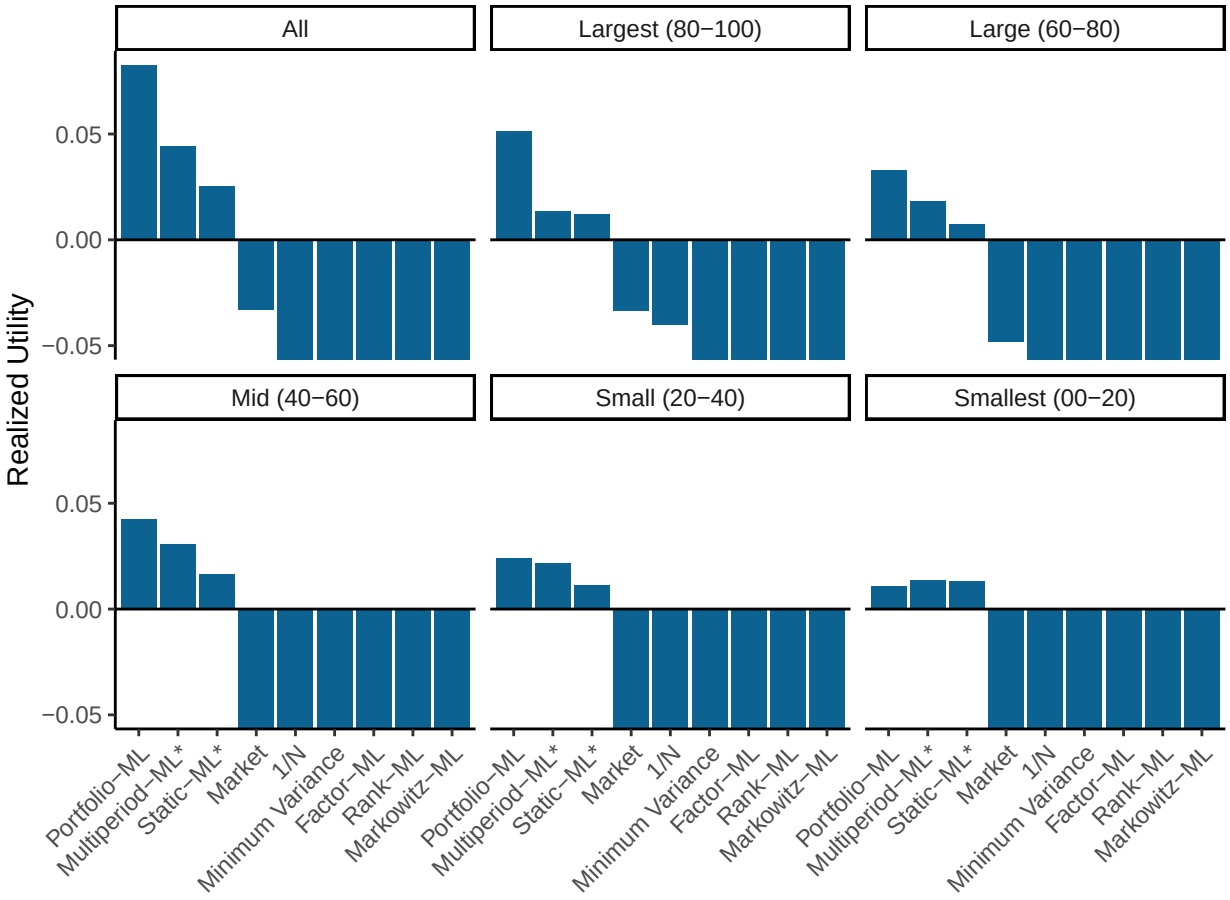

Figure 8: Performance across the size distribution

Note: The figure shows the realized utility after trading cost for different portfolio choice methods when implemented on all NYSE stocks or when the NYSE stocks are split into five size groups. The five size groups start with the largest stocks in the 80th to 100th percentile of the size distribution and end with the smallest stocks in the 00th to 20th percentile. The realized utility is computed for an investor with a wealth of $10B by the end of 2020 and a relative risk aversion of 10.

7 Conclusion

We develop a bridge between ML and portfolio choice with trading costs. To accomplish this bridge, we solve the optimal portfolio problem with transaction costs when returns are predictable via an *arbitrary* function of security characteristics and then show how the solution can be computed in a tractable way via machine learning.

To evaluate the usefulness of our method — and, in fact, any method of portfolio choice — we propose that investors and researchers should focus on the implementable efficient frontier. That is, investors should focus on out-of-sample performance net of trading costs, not the in-sample cost-agnostic performance that form the basis of the standard textbook

efficient frontier.

We show empirically that our method expands the implementable efficient frontier relative to other methods of portfolio choice. In other words, we find significant out-of-sample net-of-cost gains from our method even relative to sophisticated and more highly parameterized alternatives. We also consider several comparative statics, showing how the implementable efficient frontier contracts for larger investors facing higher market impact costs.

The method implies a novel view of which securities and characteristics are important, depending on the investor's size. Indeed, while standard methods that ignore transaction costs focus on transient features that work well on paper for small stocks, our method naturally selects persistent features of economic importance.

Future research should study the equilibrium implications of trading costs in a dynamic economy. For example, how do noise traders move prices depending on the persistence of their demand and liquidity of the affected stocks, and how do different types of arbitrageurs exploit the opportunities that arise? Our model implies that small sophisticated investors focus on high-turnover strategies among illiquid stocks while large arbitrageurs focus on more persistent signals among large stocks. This finding resembles how high-frequency traders and other proprietary trading firms remain relatively small (not accepting money from outside investors) while successful asset managers often grow large.

References

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

Balduzzi, P. and A. W. Lynch (1999). Transaction costs and predictability: some utility cost calculations. *Journal of Financial Economics 52*(1), 47–78.

Bali, T. G., H. Beckmeyer, M. Moerke, and F. Weigert (2021). Option return predictability with machine learning and big data. *Available at SSRN 3895984*.

Bali, T. G., A. Goyal, D. Huang, F. Jiang, and Q. Wen (2022). Predicting corporate bond returns: Merton meets machine learning. *Georgetown McDonough School of Business Research Paper* (3686164), 20–110.

Beaglehole, D., M. Belkin, and P. Pandit (2023). On the inconsistency of kernel ridgeless regression in fixed dimensions. *SIAM Journal on Mathematics of Data Science 5*(4), 854–872.

Brandt, M. W., P. Santa-Clara, and R. Valkanov (2009). Parametric portfolio policies: Exploiting characteristics in the cross-section of equity returns. *The Review of Financial Studies 22*(9), 3411–3447.

Breiman, L. (2001). Random forests. *Machine learning 45*(1), 5–32.

Cakici, N. and A. Zaremba (2022). Empirical asset pricing via machine learning: The global edition. *Available at SSRN 4028525*.

Chen, A. Y. and M. Velikov (2021). Zeroing in on the expected returns of anomalies. *Working paper, Federal Reserve Board*.

Chen, L., M. Pelger, and J. Zhu (2023). Deep learning in asset pricing. *Management Science*.

Choi, D., W. Jiang, and C. Zhang (2021). Alpha go everywhere: Machine learning and international stock returns. *Available at SSRN 3489679*.

Collin-Dufresne, P., K. Daniel, and M. Sağlam (2020). Liquidity regimes and optimal dynamic asset allocation. *Journal of Financial Economics 136*(2), 379–406.

Constantinides, G. M. (1986). Capital market equilibrium with transaction costs. *Journal of Political Economy 94*, 842–862.

Davis, M. and A. Norman (1990). Portfolio selection with transaction costs. *Mathematics of Operations Research 15*, 676–713.

DeMiguel, V., L. Garlappi, and R. Uppal (2009). Optimal versus naive diversification: How inefficient is the 1/n portfolio strategy? *The review of Financial studies 22*(5), 1915–1953.

DeMiguel, V., A. Martin-Utrera, F. J. Nogales, and R. Uppal (2020). A transaction-cost perspective on the multitude of firm characteristics. *The Review of Financial Studies 33*(5), 2180–2222.

Detzel, A. L., R. Novy-Marx, and M. Velikov (2021). Model selection with transaction costs. *Available at SSRN 3805379*.

Fama, E. F. and K. R. French (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics 33*(1), 3–56.

Frazzini, A., R. Israel, and T. J. Moskowitz (2018). Trading costs. *Available at SSRN 3229719*.

Freyberger, J., A. Neuhierl, and M. Weber (2020). Dissecting characteristics nonparametrically. *The Review of Financial Studies 33*(5), 2326–2377.

Gârleanu, N. and L. H. Pedersen (2013). Dynamic trading with predictable returns and transaction costs. *The Journal of Finance 68*(6), 2309–2340.

Gârleanu, N. and L. H. Pedersen (2016). Dynamic portfolio choice with frictions. *Journal of Economic Theory 165*, 487–516.

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

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

Han, Y., A. He, D. Rapach, and G. Zhou (2021). Expected stock returns and firm characteristics: E-lasso, assessment, and implications. *Assessment, and Implications (September 10, 2021)*.

Hansen, L. P. and R. Jagannathan (1991). Implications of security market data for models of dynamic economies. *Journal of political economy 99*(2), 225–262.

Heston, S. L. and R. Sadka (2008). Seasonality in the cross-section of stock returns. *Journal of Financial Economics 87*(2), 418–445.

Jensen, T. I., B. Kelly, and L. H. Pedersen (2023). Is there a replication crisis in finance? *The Journal of Finance 78*(5), 2465–2518.

Kelly, B. T., S. Malamud, and K. Zhou (2022). The virtue of complexity in return prediction.

Kelly, B. T., D. Palhares, and S. Pruitt (2022). Modeling corporate bond returns. *Journal of Finance*.

Kelly, B. T., S. Pruitt, and Y. Su (2019). Characteristics are covariances: A unified model of risk and return. *Journal of Financial Economics 134*(3), 501–524.

Koijen, R. S. and M. Yogo (2019). A demand system approach to asset pricing. *Journal of Political Economy 127*(4), 1475–1515.

Krein, M. G. and M. Rutman (1950). *Linear operators leaving invariant a cone in a Banach space*. Number 26. American Mathematical Society.

Leippold, M., Q. Wang, and W. Zhou (2022). Machine learning in the chinese stock market. *Journal of Financial Economics 145*(2), 64–82.

Li, S. A., V. DeMiguel, and A. Martin-Utrera (2020). Which factors with price-impact costs? *Victor and Martin-Utrera, Alberto, Which Factors with Price-Impact Costs*.

Liu, F., Z. Liao, and J. Suykens (2021). Kernel regression in high dimensions: Refined analysis beyond double descent. In *International Conference on Artificial Intelligence and Statistics*, pp. 649–657. PMLR.

Lynch, A. W. and P. Balduzzi (2000). Predictability and transaction costs: the impact on rebalancing rules and behavior. *Journal of Finance 55*, 2285–2309.

Maddalena, E. T., P. Scharnhorst, and C. N. Jones (2021). Deterministic error bounds for kernel-based learning techniques under bounded noise. *Automatica 134*, 109896.

Manchero, J., D. Orr, and J. Wang (2011). The barra us equity model (use4). *MSCI BARRA methodology paper*.

Markowitz, H. M. (1952). Portfolio selection. *The Journal of Finance 7*(1), 77–91.

Markowitz, H. M. (1959). *Portfolio Selection: Efficient Diversification of Investments*. Yale University Press.

Mei, S., T. Misiakiewicz, and A. Montanari (2022). Generalization error of random feature and kernel methods: hypercontractivity and kernel matrix concentration. *Applied and Computational Harmonic Analysis 59*, 3–84.

Mohri, M., A. Rostamizadeh, and A. Talwalkar (2018). *Foundations of machine learning*. MIT press.

Molnar, C. (2022). *Interpretable Machine Learning* (2 ed.).

MSCI Barra (2007). Barra Risk Model Handbook.

Muravyev, D., N. D. Pearson, and J. M. Pollet (2022). Anomalies and their short-sale costs. *Available at SSRN 4266059*.

Rahimi, A. and B. Recht (2007). Random features for large-scale kernel machines. *Advances in neural information processing systems 20*.

Sutherland, D. J. and J. Schneider (2015). On the error of random fourier features. *arXiv preprint arXiv:1506.02785*.

Van Binsbergen, J. H. and C. C. Opp (2019). Real anomalies. *The Journal of finance 74*(4), 1659–1706.

Appendix

The appendix is organized as follows. Appendix B presents implementation details, including how to compute the discount factor m (section B.1) and details on Multiperiod-ML (B.2).

Appendix C contains proofs, including a key technical lemma for verifying optimality of policies (C.2), properties of m used in the proofs (C.3), proofs of Propositions 2 and 3 (C.4), proofs of Propositions 6 and 7 (C.5), the optimality of Portfolio ML (C.6), the universal approximation properties of random feature (C.7), and economic feature importance (C.8).

Appendix D contains further empirical information, including an overview of the security characteristics used empirically (D.1), the estimated hyper-parameters over time (D.2), the autocorrelation of the features and their importance for different return prediction horizons (D.3), and various portfolio statistics over time (D.4).

Appendix E contains simulation evidence on how Portfolio-ML, Multiperiod-ML, and Static-ML perform when varying the speed of the input signals and the wealth of the investor (or, equivalently, the liquidity of the stocks traded).

A On the Economics of the Optimal Strategy

The model solution in Proposition 3 leads to several economically intuitive properties, as shown next.

Proposition 6 (Trading speed) *The matrix m is monotone decreasing in G, Σ and γ and increasing in w, in the sense of positive semi-definite order.*[21]

To understand the intuition behind these results, recall from (20) that m is the persistence of the optimal portfolio, or, equivalently, $I - m$ is the trading speed toward the aim. At the same time, m also determines how much the aim portfolio weights near-time performance versus long-term returns, as seen in (21). So, consider what happens when we move to the right in the implementable efficient frontier in Figure 1 by decreasing risk aversion. This decreasing risk aversion means that m increases, thus reducing trading speed and making the aim more focused on persistent signals. In other words, trading costs increase as the investor takes more risk, but the investor compensates by trading more slowly toward a more stable aim. Likewise, an investor with larger wealth w has a lower trading speed because of more significant market impact costs, providing economic intuition for Figure 1.B.

The next proposition considers limiting portfolios with small or large wealth.

Proposition 7 (Small and large investors) *When wealth approaches zero such that transaction costs become negligible, $w \to 0$, the discount factor converges as $m \to 0$ and the optimal portfolio policy converges to the Markowitz portfolio, $\pi_t \to \frac{1}{\gamma}\Sigma^{-1}\mu_t$.*

When wealth grows large, $w \to \infty$, the optimal portfolio diminishes, $\pi_t \to 0$, but the discount factor m and rescaled portfolio, $w\pi_t$, and aim portfolio, wA_t, converge to finite limits if Λ is diagonal and $\bar{g}_i = \frac{1+r^f+\mu_i}{1+g^w} > 1$ for all i.

[21] Given two symmetric matrices A, B, we write $A \geq B$ in the sense of positive semi-definite order if $A - B$ is positive semi-definite.

Naturally, a tiny investor holds a portfolio close to the Markowitz portfolio because of the low market impact costs. The limiting behavior as wealth goes to infinity is less obvious: As wealth grows infinite, the investor ultimately holds almost all wealth in the risk-free asset as trading a meaningful proportion of wealth in illiquid assets becomes too costly. However, this result does not mean that the portfolio in dollar terms is not large. Instead, what happens is that the portfolio, measured in dollar terms, grows toward a finite limit. In other words, a maximum amount of money can be made in the market, and as wealth increases, the investors ultimately hold this "maximum dollar portfolio."

We can also get economic insight into the optimal trading strategy by considering special cases in which the solution is especially simple. In particular, the solution can be written explicitly if we make additional assumptions on the return predictability, $E_t(r_{t+1}) = \mu_t$, as in Gârleanu and Pedersen (2013). In particular, we consider a case in which expected returns are linear in the signals, $S_t = \text{vec}(s_t) \in \mathbb{R}^{KN\times 1}$,

$$\mu_t = BS_t \tag{A.1}$$

where $B \in \mathbb{R}^{N\times KN}$ contains the predictive relations. Further, we consider signals with auto-regressive dynamics,

$$E_t(S_{t+1} - S_t) = -\Phi S_t, \tag{A.2}$$

where $\Phi \in \mathbb{R}^{KN\times KN}$ captures alpha decay. Further simplifications can be achieved when the parameters satisfy

$$\Lambda = \lambda \times \Sigma \tag{A.3}$$
$$\bar{g} = g \times I \tag{A.4}$$
$$G = g_2 \times 11' \tag{A.5}$$

where $\lambda, g, g_2 \in \mathbb{R}$. These conditions yield a simple explicit solution for the aim portfolio as shown next, where we recall that $c = \frac{\gamma}{w} m\Lambda^{-1}\Sigma$.

Proposition 8 (Simple Explicit Solution) *Under (A.1)–(A.2), the aim portfolio is*[22]

$$A_t \;=\; (I-m)^{-1}(I-(m\Lambda^{-1}\bar{g}\Lambda)\otimes(I-\Phi))^{-1}(c\gamma^{-1}\Sigma^{-1}B)S_t\,. \tag{A.6}$$

If (A.3)–(A.5) also hold, the aim is

$$A_t \;=\; \gamma^{-1}\Sigma^{-1}B\frac{m\gamma}{(1-m)\lambda w}(I-(mg)(I-\Phi))^{-1}S_t \tag{A.7}$$

where

$$m \;=\; \frac{(w^{-1}\gamma\lambda^{-1}+1+g_2)-\sqrt{(w^{-1}\gamma\lambda^{-1}+1+g_2)^2-4g_2}}{2g_2}\,. \tag{A.8}$$

and, if $\Phi = \text{diag}(\phi_i)$, the aim is the Markowitz portfolio where the signals are scaled down,

[22] Given two matrices $A \in \mathbb{R}^{N\times N}$ and $B \in \mathbb{R}^{(KN)\times(KN)}$, we can define $A\otimes B$ as a linear map on $\mathbb{R}^{N\times(KN)}$, where, for any $X \in \mathbb{R}^{N\times(KN)}$, we have $(A\otimes B)(X) = AXB \in \mathbb{R}^{N\times(KN)}$.

$\tilde{S}_i = \frac{m\gamma}{(1-m)\lambda w(1-mg+mg\phi_i)} S_i$, *depending on their alpha decay,* ϕ_i,

$$A_t \;=\; \gamma^{-1}\Sigma^{-1}B\tilde{S}_t \tag{A.9}$$

This proposition replaces the infinite sum from Proposition 3 with an explicit expression, which is particularly simple in equation (A.9). Indeed, (A.9) has exactly the same form as the Markowitz portfolio from (32), $\pi_t^{\text{Markowitz}} = \frac{1}{\gamma}\Sigma^{-1}\mu_t = \frac{1}{\gamma}\Sigma^{-1}BS_t$, except that S_t is replaced with $\tilde{S}_t$. Intuitively, $\tilde{S}_t$ has the property that a signal is used less if it has a larger alpha decay, ϕ_i.

Proof of Proposition 8. Under (A.1)–(A.2), the aim portfolio is

$$A_t \;=\; (I-m)^{-1}\sum_{\tau=0}^{\infty}(m\Lambda^{-1}\bar{g}\Lambda)^{\tau}\, c\gamma^{-1}\Sigma^{-1}B(I-\Phi)^{\tau}S_t, \tag{A.10}$$

which can be rewritten as (A.6).

To show (A.7), note that the equation for $\tilde{m}$ takes the form

$$\tilde{m} \;=\; \left(w^{-1}\gamma\Lambda^{-1/2}\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}(G\circ(\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2}))\Lambda^{-1/2}\right)^{-1}, \tag{A.11}$$

which under under (A.3) and (A.5) can be written as

$$\tilde{m} \;=\; \left(w^{-1}\gamma\lambda^{-1} + I + \Lambda^{-1/2}g_2((\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2}))\Lambda^{-1/2}\right)^{-1}. \tag{A.12}$$

Since we know that this equation has a unique solution in $\mathcal{S}(0,1)$, we can focus on the solution when $\tilde{m}$ is proportional to I. In this case, each diagonal element of $\tilde{m}$ solves

$$\tilde{m} \;=\; \left(w^{-1}\gamma\lambda^{-1} + 1 + g_2(I-\tilde{m})\right)^{-1}. \tag{A.13}$$

that is

$$g_2\tilde{m}^2 - (w^{-1}\gamma\lambda^{-1} + 1 + g_2)\tilde{m} \;+\; 1 \;=\; 0 \tag{A.14}$$

which has a unique solution in $(0,1)$, given by

$$\tilde{m} \;=\; \frac{(w^{-1}\gamma\lambda^{-1} + 1 + g_2) - \sqrt{(w^{-1}\gamma\lambda^{-1} + 1 + g_2)^2 - 4g_2}}{2g_2} \tag{A.15}$$

and $m = \Lambda^{-1/2}\tilde{m}I\Lambda^{1/2} = \tilde{m}I$. We then insert this m in (A.10), use (A.4), and simplify via geometric series to yield (A.7). Finally, (A.9) follows directly under the diagonal assumption. □

B Implementation Details

B.1 Computing the Discount Factor m

Lemma 2 *Suppose that Λ and Σ are both diagonal. Let also $\Lambda^{-1/2}\Sigma\Lambda^{-1/2} = \text{diag}(q_{i,i})$. Then there exists a unique diagonal solution $\tilde{m} \in \mathcal{S}(0,1)$ to*

$$\tilde{m} \;=\; \left(w^{-1}\gamma\Lambda^{-1/2}\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}(G \circ (\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2}))\Lambda^{-1/2}\right)^{-1}, \tag{B.1}$$

such that $\Lambda^{-1/2}\tilde{m}\Lambda^{1/2}\bar{g}$ has all eigenvalues below one in absolute value. It is given by $\tilde{m} = \text{diag}(\tilde{m}_{i,i})$, with

$$\tilde{m}_{i,i} = m_{i,i} \;=\; \frac{2}{w^{-1}\gamma q_{i,i} + G_{i,i} + 1 + \sqrt{(w^{-1}\gamma q_{i,i} + G_{i,i} + 1)^2 - 4G_{i,i}}} \tag{B.2}$$

Proof of Lemma 2. The fact that (B.2) solves (B.1) follows by direct calculation. Indeed, we get

$$\tilde{m}_{i,i} \;=\; \frac{1}{w^{-1}\gamma q_{i,i} + 1 + G_{i,i} - G_{i,i}\tilde{m}_{i,i}}, \tag{B.3}$$

that is

$$(w^{-1}\gamma q_{i,i} + 1 + G_{i,i} - G_{i,i}\tilde{m}_{i,i})\tilde{m}_{i,i} \;=\; 1, \tag{B.4}$$

that is

$$G_{i,i}\tilde{m}_{i,i}^2 \;-(w^{-1}\gamma q_{i,i} + 1 + G_{i,i})\tilde{m}_{i,i} \;+\; 1 \;=\; 0, \tag{B.5}$$

that is, using the identity $\sqrt{a} - \sqrt{b} = \frac{a-b}{\sqrt{a}+\sqrt{b}}$, we get

$$\begin{aligned}
\tilde{m}_{i,i} \;&=\; \frac{(w^{-1}\gamma q_{i,i} + 1 + G_{i,i}) \pm \sqrt{(w^{-1}\gamma q_{i,i} + 1 + G_{i,i})^2 - 4G_{i,i}}}{2G_{i,i}} \\
&= \frac{4G_{i,i}}{2G_{i,i}((w^{-1}\gamma q_{i,i} + 1 + G_{i,i}) \mp \sqrt{(w^{-1}\gamma q_{i,i} + 1 + G_{i,i})^2 - 4G_{i,i}})} \\
&= \frac{2}{(w^{-1}\gamma q_{i,i} + 1 + G_{i,i}) \mp \sqrt{(w^{-1}\gamma q_{i,i} + 1 + G_{i,i})^2 - 4G_{i,i}}}
\end{aligned} \tag{B.6}$$

Furthermore, $\Lambda^{-1/2}\tilde{m}\Lambda^{1/2}\bar{g} = \text{diag}(\tilde{m}_{i,i}\bar{g}_i)$ because all matrices involved are diagonal. The last claim follows because $G = \bar{g}\bar{g}' + \frac{1}{(1+r^f+\bar{\mu})^2}\Sigma$ implies that $G_{i,i} \geq \bar{g}_i^2$ and, hence, since, $g_i^2 + 1 \geq 2g_i$, we get for $\tilde{m}_{i,i}(-)$ that

$$\tilde{m}_{i,i}(-) \;<\; \frac{2}{G_{i,i}+1} \;\leq\; \frac{2}{g_i^2+1} \;\leq\; g_i^{-1} \tag{B.7}$$

By contrast,

$$\begin{aligned}\tilde{m}_{i,i}(+) &= \frac{(w^{-1}\gamma q_{i,i}+1+G_{i,i})+\sqrt{(w^{-1}\gamma q_{i,i}+1+G_{i,i})^2-4G_{i,i}}}{2G_{i,i}} \\ &> \frac{(1+G_{i,i})+|G_{i,i}-1|}{2G_{i,i}}\,.\end{aligned} \tag{B.8}$$

If $G_{i,i}>1$, we get

$$\tilde{m}_{i,i}(+) > 1 \tag{B.9}$$

so that $\tilde{m}\notin\mathcal{S}(0,1)$. If $G_{i,i}<1$, then

$$\tilde{m}_{i,i}(+) > 1/G_{i,i} > 1, \tag{B.10}$$

so that, once again, $\tilde{m}\notin\mathcal{S}(0,1)$. Thus, $\tilde{m}(-)$ is the only diagonal solution in $\mathcal{S}(0,1)$. □

Another special case with a closed-form solution is when G has a rank of one, which is not a realistic case but is a useful approximation. Specifically, we have that $G=\bar{g}\bar{g}'+\frac{1}{(1+r^f+\bar{\mu})^2}\Sigma\cong\bar{g}\bar{g}'$, where[23] $\bar{g} = (1+r^f+\bar{\mu})^{-1}(1+r^f+E[\mu])$ and the approximation is based on the idea that $\bar{g}$ is a vector of numbers close to one, whereas Σ is much smaller with numbers of the order of 0.10^2 when monthly volatility is around 10%. The following lemma derives a closed-form solution to (14) from the main text for the case when $G=\xi\xi'$ for some vector $\xi>0$.

Lemma 3 *Suppose that Λ is diagonal. In the case when $G=\xi\xi'$ for some vector $\xi>0$, the unique solution $\tilde{m}=\tilde{m}(\xi)$ to (14) is given by*

$$\tilde{m} = \operatorname{diag}(\xi)^{-1/2}\,0.5(\hat{\Sigma}-\left(\hat{\Sigma}^2-4I\right)^{1/2})\operatorname{diag}(\xi)^{-1/2}\,, \tag{B.11}$$

and $\hat{\Sigma} = \operatorname{diag}(\xi)^{-1/2}(w^{-1}\Lambda^{-1/2}\gamma\Sigma\Lambda^{-1/2}+\operatorname{diag}((\xi_i^2+1)))\operatorname{diag}(\xi)^{-1/2}$ and $\left(\hat{\Sigma}^2-4I\right)^{1/2}$ is the unique positive-definite square root. Furthermore, $m=\Lambda^{-1/2}\tilde{m}\Lambda^{1/2}$ is such that $m\operatorname{diag}(\xi)$ has all eigenvalues below one in absolute value.

Proof of Lemma 3. We normalize $\gamma=w=1$, which is without loss of generality since we can absorb these into Σ. Recall that (14) is

$$\tilde{m} = \left(\Lambda^{-1/2}\Sigma\Lambda^{-1/2}+I+\Lambda^{-1/2}\operatorname{diag}(\xi)(\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2})\operatorname{diag}(\xi)\Lambda^{-1/2}\right)^{-1}, \tag{B.12}$$

because $G\circ A=(\xi\xi')\circ A=\operatorname{diag}(\xi)A\operatorname{diag}(\xi)$ for any matrix A. The assumption of a diagonal Λ implies

$$\tilde{m} = \left(\Lambda^{-1/2}\Sigma\Lambda^{-1/2}+I+\operatorname{diag}(\xi)(I-\tilde{m})\operatorname{diag}(\xi)\right)^{-1}. \tag{B.13}$$

We abuse the notation and use ξ to denote $\operatorname{diag}(\xi)$. Let $\tilde{\Sigma} = \Lambda^{-1/2}\Sigma\Lambda^{-1/2}+I+\xi^2$. Then,

[23] We abuse the notation and use $\bar{g}$ to denote both the vector and the diagonal matrix.

(B.13) takes the form

$$\tilde{m} \;=\; \left(\tilde{\Sigma} - \xi\tilde{m}\xi\right)^{-1}. \tag{B.14}$$

Define

$$\hat{\Sigma} \;=\; \xi^{-1/2}\tilde{\Sigma}\xi^{-1/2} \;=\; \xi^{-1/2}\Lambda^{-1/2}\Sigma\Lambda^{-1/2}\xi^{-1/2} + \xi^{-1} + \xi \;>\; 2I\,,$$

where the last inequality follows because $\xi + \xi^{-1} \geq 2$ for any positive number ξ. Let also $\hat{m} = \xi^{1/2}\tilde{m}\xi^{1/2}$. Then, we get

$$\hat{m}^2 - \hat{\Sigma}\hat{m} \;+\; I \;=\; 0\,. \tag{B.15}$$

Let UDU' be the eigendecomposition of $\hat{\Sigma}$, and $\tilde{m} = U'\hat{m}U$. Then, multiplying this identity by U and U' from the left- and right-hand sides, respectively, we get

$$\tilde{m}^2 - D\tilde{m} + I \;=\; 0\,. \tag{B.16}$$

If $\tilde{m}$ is degenerate, then $\tilde{m}h = 0$ for some h, and hence so is $\tilde{m}^2h - D\tilde{m}h + h = 0$ implies $h = 0$. Thus, $\tilde{m}$ is non-degenerate. Multiplying by $\tilde{m}^{-1}$, we get

$$D \;=\; \tilde{m} + \tilde{m}^{-1}\,.$$

Let $\tilde{m} = VdV'$ be the eigen-decomposition of $\tilde{m}$. Then,

$$V'DV \;=\; d + d^{-1},$$

implying that $V'DV$ is diagonal. Since D itself is diagonal and D and VDV' have identical eigenvalues, it must be that $V'DV$ and D are identical up to a permutation. Thus, without loss of generality, we may assume that $V'DV = D$ and get

$$D \;=\; d + d^{-1}\,.$$

This is a system of N quadratic equations $D_i = d_i + d_i^{-1}$. This system has 2^N solutions, $d_i = 0.5(D_i \pm \sqrt{D_i^2 - 4})$. Since $D_i \geq 2$, we have that only the solution $d_i = 0.5(D_i - \sqrt{D_i^2 - 4})$ satisfies $d_i \leq 1$. Thus, the unique solution $\hat{m}$ in $\mathcal{S}(0,1)$ satisfies $d = 0.5(D - \sqrt{D^2 - 4})$. The diagonal matrix D commutes with V if and only if V has a block structure, with each block corresponding to a different eigenvalue of D. As a result, V also commutes with d, so that $VdV' = d$. Thus,

$$\hat{m} \;=\; UVdV'U' \;=\; UdU' \;=\; 0.5(\hat{\Sigma} - (\hat{\Sigma}^2 - 4I)^{1/2})\,. \tag{B.17}$$

□

Starting with this approximation, we can compute the exact m stepwise, as follows. Note first that the set $\mathcal{S}$ of symmetric, positive definite matrices is a partially-ordered set with respect to the positive semi-definite order: We say that $m_1 \leq m_2$ if $m_2 - m_1$ is positive semi-definite. Further, we let $\mathcal{S}(0,1)$ be the set of positive semi-definite matrices with eigenvalues between zero and one. Below, we show that (B.1) always have a unique solution $\tilde{m} \in \mathcal{S}(0,1)$. The following lemma shows how to construct this unique solution.

Lemma 4 *Define*

$$F(\tilde{m}) \;=\; \left(w^{-1}\Lambda^{-1/2}\gamma\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}((\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2})\circ G)\Lambda^{-1/2}\right)^{-1}. \tag{B.18}$$

The unique solution $\tilde{m} \in \mathcal{S}(0,1)$ to $\tilde{m} = F(\tilde{m})$ can be computed by iterating map F. Indeed, F maps $\mathcal{S}(0,1)$ into itself and is monotonic with respect to the positive semi-definite order. Furthermore, it has a unique fixed point $\tilde{m}_* \in \mathcal{S}(0,1)$, and its iterations converge to this unique fixed point from any starting point m_0 in $\mathcal{S}(0,1)$ satisfying either $m_0 \leq F(m_0)$ or $m_0 \geq F(m_0)$. In particular, it converges upward from the smallest starting point $0 \in \mathcal{S}(0,1)$:

$$F(0) \leq F(F(0)) \;\leq\; \cdots \;\leq\; F(\cdots(F(0))) \;\to\; \tilde{m}_*\,. \tag{B.19}$$

Furthermore, the map F is monotone decreasing in the matrix G in the sense of positive semi-definite order: $\tilde{m}(G_1) \;\leq\; \tilde{m}(G_2)$ whenever $G_1 \geq G_2$. In particular, if $G = \bar{g}\bar{g}' + \frac{1}{(1+r^f+\bar{\mu})^2}\Sigma$ with $\bar{g} \;=\; (1+r^f+\bar{\mu})^{-1}(1+r^f+E[\mu])$, and if Λ is diagonal, then it also converges downward from the starting point $\tilde{m}(\bar{g})$ from Lemma 3:

$$F(\tilde{m}(\bar{g})) \geq F(F(\tilde{m}(\bar{g}))) \;\geq\; \cdots \;\geq\; F(\cdots(F(\tilde{m}(\bar{g})))) \;\to\; \tilde{m}_*. \tag{B.20}$$

The first terms in (B.19) and (B.20) provide lower and upper bounds for the solution $\tilde{m}_*$:

$$\begin{aligned}&\left(w^{-1}\Lambda^{-1/2}\gamma\Sigma\Lambda^{-1/2} + \mathrm{diag}((G_{i,i}+1))\right)^{-1}\\ &\leq \tilde{m}_* \;\leq\; \left(w^{-1}\Lambda^{-1/2}\gamma\Sigma\Lambda^{-1/2} + I + ((I-\tilde{m}(\bar{g}))\circ G)\right)^{-1}.\end{aligned} \tag{B.21}$$

Proof. The only claim that requires proof is the fact that $\tilde{m}(\bar{g}) \;\geq\; F(\tilde{m}(\bar{g}); G)$. Indeed, by the monotonicity of F in G and the fact that $G \geq \bar{g}\bar{g}'$, we have

$$\tilde{m}(\bar{g}) \;=\; F(\tilde{m}(\bar{g}); \bar{g}\bar{g}') \;\geq\; F(\tilde{m}(\bar{g}); G)\,,$$

and the claim follows. This inequality, combined with monotonicity, implies the required sequence of inequalities (B.20). □

In the empirical analysis, we compute m by starting from the approximation in (B.17), and then iterate ten times on the map in (B.18).

B.2 Implementation Details for Multiperiod-ML

Suppose that Λ is diagonal. While formula (21) requires an infinite sum, in our numerical implementation, we use the approximation

$$\begin{aligned}
A_t &= (I-m)^{-1}(I-(m\,\mathrm{diag}(\bar{g})))^2(I-(m\,\mathrm{diag}(\bar{g})))^{-2}\sum_{\tau=0}^{\infty}(m\,\mathrm{diag}(\bar{g}))^{\tau}c\Sigma^{-1}E_t[r_{t+1+\tau}] \\
&= (I-m)^{-1}(I-(m\,\mathrm{diag}(\bar{g})))^2\sum_{\tau=0}^{\infty}(m\,\mathrm{diag}(\bar{g}))^{\tau}(I-(m\,\mathrm{diag}(\bar{g})))^{-2}c\Sigma^{-1}E_t[r_{t+1+\tau}] \\
&= (I-m)^{-1}(I-(m\,\mathrm{diag}(\bar{g})))(I-(m\,\mathrm{diag}(\bar{g})))\sum_{\tau=0}^{\infty}(m\,\mathrm{diag}(\bar{g}))^{\tau}\,\widetilde{c}\,\Sigma^{-1}E_t[r_{t+1+\tau}] \\
&\approx (I-m)^{-1}(I-(m\,\mathrm{diag}(\bar{g})))^2(I-(m\,\mathrm{diag}(\bar{g}))^{k+1})^{-1}\sum_{\tau=0}^{k}(m\,\mathrm{diag}(\bar{g}))^{\tau}\,\widetilde{c}\,\Sigma^{-1}E_t[r_{t+1+\tau}]
\end{aligned} \tag{B.22}$$

where we have defined

$$\widetilde{c} = (I-(m\,\mathrm{diag}(\bar{g})))^{-2}c\,. \tag{B.23}$$

C Proofs

C.1 Properties of the Implementable Efficient Frontier

We note that the textbook efficient frontier is usually defined in terms of a risk-minimization problem rather than the return-maximization in (12). Hence, we could consider the corresponding definition of the implementable efficient frontier as the combination of volatilities and expected net returns, $(\sigma(k),k)_{k\geq 0}$, such that risk is minimal for that level of net return:

$$\sigma(k)^2 = \min_{\pi_t} E\left[\pi_t'\Sigma\pi_t\right] \quad \text{s.t. } E\left[r_{t+1}^{\pi,net}\right] = k \tag{C.1}$$

However, this definition is less helpful for two reasons. First, no solution exists for large enough k. Second, this definition cannot produce the downward-sloping portion of the frontier seen in Figure 1.

Proof of Proposition 1. (i) We first show that the net Sharpe ratio is decreasing along the frontier. To see that, consider two risk levels, $\sigma_1 < \sigma_2$. Let π_t^2 be the frontier portfolio corresponding to σ_2. This portfolio has the highest expected net return for this risk level and the highest net Sharpe ratio. If we scale down this portfolio to $\pi_t^1 = \frac{\sigma_1}{\sigma_2}\pi_t^2$ (putting the rest of the money in the risk-free asset), then the risk becomes $\frac{\sigma_1}{\sigma_2}\sigma^2 = \sigma_1$. Then we have:

$$\max_{\pi_t\in\Pi \text{ s.t. } E[\pi_t'\Sigma\pi_t]=\sigma_1^2} E\left[r_{t+1}^{\pi,net}\right] \geq E\left[r_{t+1}^{\pi_t^1,net}\right] > \frac{\sigma_1}{\sigma_2}E\left[r_{t+1}^{\pi_t^2,net}\right] \tag{C.2}$$

Here, the first inequality follows from the definition of the frontier as the maximum. The second inequality follows from the fact that gross returns are linear, but transaction costs

are quadratic and $\frac{\sigma_1}{\sigma_2} < 1$. Dividing both sides of (C.7) by σ_1, we see that the net Sharpe ratio is decreasing in σ.

(vi) Consider the implementable efficient frontier corresponding to wealth levels w_1 and w_2, where $w_1 < w_2$. Take a point on the frontier of w_2 corresponding to the risk of σ and a portfolio π_t^2. Then, with wealth w_1, the portfolio π_t^2 delivers a higher net return with the same risk (and there may exist another portfolio with an even higher net return for this level of wealth), so the frontier of w_1 must be above that of w_2.

(ii)–(iii) To prove the concavity of the implementable efficient frontier, consider two risk levels, $\sigma_1,\ \sigma_2$. Let $\pi(\sigma)$ be the frontier portfolio corresponding to σ, and let

$$R(\sigma) \ = \ \max_{\pi_t \in \Pi \text{ s.t. } E[\pi_t' \Sigma \pi_t] = \sigma^2} E\left[r_{t+1}^{\pi,net}\right] \ . \tag{C.3}$$

Define

$$\tilde{\pi} \ = \ 0.5(\pi(\sigma_1) + \pi(\sigma_2)) \, . \tag{C.4}$$

Then,

$$\begin{aligned} E[\tilde{\pi}' \Sigma \tilde{\pi}] \ &= \ E[0.25((\pi(\sigma_1)' \Sigma \pi(\sigma_1) + (\pi(\sigma_2)' \Sigma \pi(\sigma_2) + 2(\pi(\sigma_1)' \Sigma \pi(\sigma_2))] \\ &\leq \ 0.25(\sigma_1^2 + \sigma_2^2 + 2\sigma_1\sigma_2) \ = \ (0.5(\sigma_1 + \sigma_2))^2 \, , \end{aligned} \tag{C.5}$$

where we have used a modified Cauchy-Schwarz inequality:[24]

$$\begin{aligned} E[(\pi(\sigma_1)' \Sigma \pi(\sigma_2)] \ &\leq \ E[((\pi(\sigma_1)' \Sigma \pi(\sigma_1))^{1/2} ((\pi(\sigma_2)' \Sigma \pi(\sigma_2))^{1/2}] \\ &\leq \ E[((\pi(\sigma_1)' \Sigma \pi(\sigma_1))]^{1/2} E[((\pi(\sigma_2)' \Sigma \pi(\sigma_2))]^{1/2} \ = \ \sigma_1 \sigma_2 \, . \end{aligned} \tag{C.6}$$

Furthermore,

$$\begin{aligned} 0.5(R(\sigma_1) + R(\sigma_2)) \ &= \ 0.5(E\left[r_{t+1}^{\pi(\sigma_1),net}\right] \ + \ E\left[r_{t+1}^{\pi(\sigma_1),net}\right]) \\ &= \ E\left[r_{t+1}' \tilde{\pi} \ - \ 0.5(TC^{\pi(\sigma_1)} + TC^{\pi(\sigma_2)})\right] \, . \end{aligned} \tag{C.7}$$

Since transaction costs are convex in π_t, we have

$$-0.5(TC^{\pi(\sigma_1)} + TC^{\pi(\sigma_2)}) \ \leq \ -TC^{\tilde{\pi}} \, .$$

Thus,

$$E\left[r_{t+1}^{\tilde{\pi},net}\right] \ \geq \ 0.5(R(\sigma_1) + R(\sigma_2)) \, ,$$

while

$$\sigma(\tilde{\pi}) \ = \ (E[\tilde{\pi}' \Sigma \tilde{\pi}])^{1/2} \ \leq \ 0.5(\sigma_1 + \sigma_2)$$

by (C.5). Thus, we get $R(\sigma(\tilde{\pi})) \geq 0.5(R(\sigma_1) + R(\sigma_2))$, and hence, the required concavity follows if $R(\sigma)$ is increasing on $[0.5(\sigma_1 + \sigma_2), \sigma(\tilde{\pi})]$.

Now, pick a $\gamma > 0$. Then, clearly, π^γ belongs to the efficient frontier, corresponding to some $\sigma(\gamma)$: Otherwise, we could increase the net expected return, keeping the variance

[24] First, $x' \Sigma y \leq x' \Sigma x y' \Sigma y$ for any positive definite Σ. Second, $E[XY] \leq E[X^2]^{1/2} E[Y^2]^{1/2}$.

fixed. We also note that for large γ, the effect of transaction costs is negligible and, hence, the efficient frontier for $\sigma \approx 0$ approximately coincides with the frictionless one, and hence $k(\sigma)$ is monotone increasing for $\sigma \approx 0$.

The set of eligible portfolios Π is a Hilbert space H, and we can define bounded, self-adjoint operators A, B and a vector $x \in H$ so that $E[r^{\pi,net}] = \langle x, \pi\rangle - 0.5\langle A\pi, \pi\rangle$ and $E[\pi'\Sigma\pi] = \langle \pi B, \pi\rangle$, where $\langle\cdot,\cdot\rangle$ is the inner product in the Hilbert space. Then, the first order condition is

$$x \; - \; A\pi \; = \; \lambda\, B\pi\,, \tag{C.8}$$

where λ is the Lagrange multiplier of the constraint $\langle \pi, B\pi\rangle = \sigma^2$. Now,

$$\pi \; = \; (A + \lambda B)^{-1} x\,, \tag{C.9}$$

and we need to solve the equation

$$\langle (A + \lambda B)^{-1} B (A + \lambda B)^{-1} x, x\rangle = \sigma^2\,. \tag{C.10}$$

For the increasing part of the frontier, the constraint $\langle \pi, B\pi\rangle \le \sigma^2$ is binding and $\lambda > 0$. This defines $\sigma_*^2 = \langle A^{-1}BA^{-1}x, x\rangle = \max_{\lambda>0}\langle (A+\lambda B)^{-1}B(A+\lambda B)^{-1}x, x\rangle$. We thus need to maximize

$$\begin{aligned} &\langle x, (A + \lambda B)^{-1} x\rangle - 0.5\langle (A + \lambda B)^{-1}(A + \lambda B - \lambda B)x, (A + \lambda B)^{-1} x\rangle \\ &\; = \; 0.5\langle x, (A + \lambda B)^{-1} x\rangle + 0.5\lambda\sigma^2\,. \end{aligned} \tag{C.11}$$

Now, for the mean-variance optimization problem

$$\max_{\pi}\{\langle x, \pi\rangle - 0.5\langle A\pi, \pi\rangle \; - \; 0.5\gamma\langle B\pi, \pi\rangle\,\},$$

and its solution is

$$\pi \; = \; (A + \gamma B)^{-1} x\,, \tag{C.12}$$

and the variance

$$0.5\gamma\langle B\pi, \pi\rangle \; = \; \langle (A + \gamma B)^{-1} x, B(A + \gamma B)^{-1} x\rangle \tag{C.13}$$

is monotone decreasing in γ and converges to σ_*^2 when $\gamma \to 0$. At the same time,

$$R(\gamma) \; = \; \langle (A + \gamma B)^{-1} x, x\rangle \tag{C.14}$$

is also monotone decreasing in γ. □

C.2 Verification Lemma

Our proofs are based on the following auxiliary result.

Lemma 5 *For simplicity, we normalize $\gamma/w = 1$. Let $\bar\Lambda_t = E_t[g_{t+1}\Lambda g_{t+1}]$. For any solution m_t to*

$$m_t \; = \; (\Sigma + \Lambda + \bar\Lambda_t)^{-1}\left(E_t[g_{t+1}\Lambda m_{t+1} g_{t+1}]m_t + \Lambda\right), \tag{C.15}$$

define

$$N_{t,t+\tau} \;=\; \prod_{\theta=0}^{\tau-1} m_{t+\theta}\, g_{t+\theta+1},\;\; N_{t,t} \;=\; I\,, \tag{C.16}$$

and

$$\tilde{N}_{t,t+\tau} \;=\; \prod_{\theta=0}^{\tau-1} m_{t+\theta}\, \Lambda^{-1} g_{t+\theta+1}\Lambda\;,\;\; \tilde{N}_{t,t} \;=\; I\,. \tag{C.17}$$

Suppose that

$$esssup \sum_{\tau} (\mathrm{tr}\; E_t[\tilde{N}'_{t,t+\tau}\tilde{N}_{t,t+\tau}])^{1/2} \;<\; \infty \tag{C.18}$$

and

$$\sum_{\tau} (\mathrm{tr}\; E[N'_{t,t+\tau}N_{t,t+\tau}])^{1/2} \;<\; \infty\,. \tag{C.19}$$

Define

$$c_t \;=\; m_t\Lambda^{-1}\Sigma \tag{C.20}$$

and

$$Q_t \;=\; E_t\left[\sum_{\tau=0}^{\infty} \tilde{N}_{t,t+\tau} c_{t+\tau} Markowitz_{t+\tau}\right] \tag{C.21}$$

and let

$$\pi_t \;=\; \sum_{\theta=0}^{\infty} N_{t-\theta,t} Q(s_{t-\theta})\,. \tag{C.22}$$

Then, all series converge in L_2 and π_t is optimal among all admissible processes π_t. Furthermore, it satisfies the recursive relationship

$$\pi_t \;=\; Q(s_t) \;+\; m_t g_t \pi_{t-1}\,. \tag{C.23}$$

We will need the following simple lemma.

Lemma 6 *We have*

$$\|Aq\| \;\le\; \|q\|\|A\| \;\le\; \|q\|\;\mathrm{tr}(A'A)^{1/2} \tag{C.24}$$

for any matrix A and any vector q.

Proof of Lemma 5. We use the ergodic theorem. Let Ω be the probability space of infinite sequences $(\cdots, s_{t-2}, s_{t-1}, s_t, s_{t+1}, \cdots)$ equipped with the standard measure. Let also $s_{t-} = (\cdots, s_{t-2}, s_{t-1})$. Any functional $\Pi(x) = \Pi(x_0, x_{-1}, x_{-2}, \cdots)$ defines an adapted, stationary trading strategy via $\pi_t = \Pi(s_t, s_{t-1}, \cdots)$. Given the time shift operator $\mathcal{T}$:

$(\cdots, s_{t-2}, s_{t-1}, s_t) \to (\cdots, s_{t-1}, s_t, s_{t+1})$, ergodicity implies that (π_t, s_t, s_{t-}) and $(\pi_{t+1}, s_{t+1}, s_t, \cdots)$ have the same distribution.

Let

$$\mathcal{O}(\pi) = E\left[-2\mu(s_t)'\pi_t + \pi_t'\Sigma\pi_t + (\pi_t - g_t\pi_{t-1})'\,\Lambda\,(\pi_t - g_t\pi_{t-1})\right] \tag{C.25}$$

Standard convexity arguments imply that it suffices to derive and verify the first-order conditions for $\min_{\pi\in\Pi}\mathcal{O}(\pi)$ for our candidate solution.

Recall that s_{t-} denotes the history of s_t and let $\pi_t = \pi(s_t, s_{t-})$ be a candidate optimal policy. Then, by direct calculation, using the ergodicity property, we get by the law of iterated expectations that

$$E[\pi_{t-1}'g_t\Lambda g_t\pi_{t-1}] \;=\; E[\pi_t'g_{t+1}\Lambda g_{t+1}\pi_t] \;=\; E[\pi_t'\hat{\Lambda}_t\pi_t]\,,$$

where

$$\bar{\Lambda}_t \;=\; E_t[g_{t+1}\Lambda g_{t+1}]\,. \tag{C.26}$$

Therefore,

$$\begin{aligned}
\mathcal{O}(\pi) \;&=\; E\left[-2\mu(s_t)'\pi_t + \pi_t'\Sigma\pi_t + (\pi_t - g_t\pi_{t-1})'\,\Lambda\,(\pi_t - g_t\pi_{t-1})\right] \\
&=\; E\Bigg[-2\,\mu(s_t)'\pi(s_t, s_{t-}) \;+\; \pi(s_t, s_{t-})'(\Sigma + \Lambda + \bar{\Lambda}_t)\pi(s_t, s_{t-})\Bigg] \\
&- 2E[\pi(s_t, s_{t-})'g_{t+1}\Lambda\pi(s_{t+1}, s_{t+1-})]\,,
\end{aligned} \tag{C.27}$$

In order to compute the first-order conditions, we need to calculate the Frechet derivative of (C.27) with respect to π. To this end, we consider a small perturbation $\pi \to \pi + \varepsilon Y$ and calculate the first order term in ε, so that

$$\mathcal{O}(\pi_t + \varepsilon Y_t) \;=\; \mathcal{O}(\pi_t) \;+\; \varepsilon\,E[\mathcal{D}(\pi)_t'\,Y_t] \;+\; O(\varepsilon^2) \tag{C.28}$$

and $\mathcal{D}(\pi)$ is the Frechet derivative. To this end, we compute

$$\begin{aligned}
&E[\mu(s_t)'(\pi(s_t, s_{t-}) + \varepsilon Y_t)] \;=\; E[\mu(s_t)'\pi(s_t, s_{t-})] + \varepsilon E[\mu(s_t)'Y_t], \\
&E[(\pi(s_t, s_{t-}) + \varepsilon Y_t)'(\Sigma + \Lambda + \bar{\Lambda}_t)(\pi(s_t, s_{t-}) + \varepsilon Y_t)] \\
&=\; E[\pi(s_t, s_{t-})'(\Sigma + \Lambda + \bar{\Lambda}_t)\pi(s_t, s_{t-})] \;+\; 2\varepsilon\,E[\pi(s_t, s_{t-})'(\Sigma + \Lambda + \bar{\Lambda})Y_t] \;+\; O(\varepsilon^2), \\
&E[(\pi(s_t, s_{t-}) + Y_t)'g_{t+1}\Lambda(\pi(s_{t+1}, s_{t+1-}) + Y_{t+1})] \\
&=\; E[\pi(s_t, s_{t-})'g_{t+1}\Lambda\pi(s_{t+1}, s_{t+1-})] \\
&+\; \varepsilon\left(E[Y_t'E_t[g_{t+1}\Lambda\pi(s_{t+1}, s_{t+1-})]] \;+\; E[\pi(s_t, s_{t-})'g_{t+1}\Lambda Y_{t+1}]\right).
\end{aligned} \tag{C.29}$$

Furthermore, by stationarity,

$$E[\pi(s_t, s_{t-})Y_{t+1}] \;=\; E[\pi(s_{t-1}, s_{t-1-})Y_t]\,.$$

We conclude that the Frechet derivative is given by

$$\begin{aligned}\mathcal{D}(\pi)_t \;&=\; -2\mu(s_t) \;+\; 2(\Sigma+\Lambda+\bar{\Lambda})\pi(s_t,s_{t-}) \\ &-\; 2E_t[g_{t+1}\Lambda\pi(s_{t+1},s_{t+1-})] \;-\; 2\Lambda g_t\pi(s_{t-1},s_{t-1-})\,,\end{aligned} \tag{C.30}$$

where $\bar{\Lambda}(s_t)=E_t[\Lambda(s_{t+1})]$, implying that π is optimal in Π if it satisfies the *integral equation*

$$\begin{aligned}&-2\mu(s_t) \;+\; 2(\Sigma+\Lambda+\bar{\Lambda})\pi(s_t,s_{t-}) \\ &-\; 2E_t[g_{t+1}\Lambda\pi(s_{t+1},s_{t+1-})] \;-\; 2\Lambda g_t\pi(s_{t-1},s_{t-1-}) \;=\; 0\,,\end{aligned} \tag{C.31}$$

which, with a bit of algebra, can be rewritten as

$$\pi(s_t,s_{t-}) \;=\; (\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Big(\mu(s_t) \;+\; E_t[g_{t+1}\Lambda\pi(s_{t+1},s_{t+1-})] \;+\; \Lambda g_t\pi(s_{t-1},s_{t-1-})\Big)\,. \tag{C.32}$$

Let us make the Ansatz

$$\pi_t \;=\; Q_t \;+\; m_t g_t \pi_{t-1}\,. \tag{C.33}$$

Then, we get

$$\pi_{t+1}=\pi(s_{t+1},s_{t+1-}) \;=\; Q_{t+1} \;+\; m_{t+1}g_{t+1}\pi_t \;=\; Q_{t+1} \;+\; m_{t+1}g_{t+1}Q_t \;+\; m_{t+1}g_{t+1}m_t g_t\pi_{t-1} \tag{C.34}$$

and, hence, (C.32) can be rewritten as

$$\begin{aligned}&Q_t \;+\; m_t g_t\pi_{t-1} \\ &=\; (\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Big(\mu(s_t) \\ &+\; E_t[g_{t+1}\Lambda\pi_{t+1}] \;+\; \Lambda g_t\pi_{t-1}\Big) \\ &=(\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Big(\mu(s_t) \\ &+\; E_t[g_{t+1}\Lambda(Q_{t+1}+m_{t+1}g_{t+1}Q_t+m_{t+1}g_{t+1}m_tg_t\pi_{t-1})] \;+\; \Lambda g_t\pi_{t-1}\Big)\,.\end{aligned} \tag{C.35}$$

Equating the coefficients on π_{t-1} gives an integral equation for m_t :

$$m_t g_t \;=\; (\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Big(E_t[g_{t+1}\Lambda m_{t+1}g_{t+1}m_tg_t]+\Lambda g_t\Big)\,, \tag{C.36}$$

and (C.35) turns into an integral equation for Q_t :

$$Q_t \;=\; (\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Big(\mu(s_t) \;+\; E_t[g_{t+1}\Lambda(Q_{t+1}+m_{t+1}g_{t+1}Q_t)]\Big)\,. \tag{C.37}$$

Dividing by g_t,, we get that (C.36) turns into the required equation (C.15). Furthermore, we can rewrite (C.36) as

$$(I-(\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}E_t[g_{t+1}\Lambda m_{t+1}g_{t+1}])m_t \;=\; (\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Lambda\,, \tag{C.38}$$

which implies

$$(I-(\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}E_t[g_{t+1}\Lambda m_{t+1}g_{t+1}]) \;=\; (\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Lambda m_t^{-1}\,. \tag{C.39}$$

After a few algebraic transformations, we get that (C.37) is equivalent to

$$(I-(\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}E_t[g_{t+1}\Lambda m_{t+1}g_{t+1}])Q_t \;=\; (\Sigma+\Lambda+\bar{\Lambda}_t)^{-1}\Big(\mu(s_t) \;+\; E_t[g_{t+1}\Lambda Q_{t+1}]\Big)\,. \tag{C.40}$$

Substituting from (C.39), we can rewrite (C.40) as

$$Q_t \;=\; m_t\Lambda^{-1}\Big(\mu(s_t) \;+\; E_t[g_{t+1}\Lambda Q_{t+1}]\Big)\,. \tag{C.41}$$

Defining

$$c_t \;=\; m_t\Lambda^{-1}\Sigma, \tag{C.42}$$

we can rewrite (C.41) as

$$Q_t \;=\; c_t\,Markowitz_t \;+\; E_t[m_t\Lambda^{-1}g_{t+1}\Lambda Q_{t+1}]\,. \tag{C.43}$$

Iterating this equation, we get the required formula up to the convergence statement. Convergence in L_2 follows directly from the made assumptions. Indeed,

$$E[E_t[X]^2] \;\leq\; E[X^2]\,, \tag{C.44}$$

and hence we can ignore $E_t[]$ when proving convergence. Let

$$q_t \;=\; c_t\,Markowitz_t\,. \tag{C.45}$$

By the made uniform boundedness assumptions and the uniform positive-definiteness of Σ, we have

$$\|q_{t+\tau}\| \;=\; \|c_{t+\tau}Markowitz_{t+\tau}\| \;\leq\; K$$

for some $K > 0$, almost surely. Since

$$\begin{aligned} \|Q_t\| &= \|E_t\left[\sum_{\tau=0}^{\infty} \tilde{N}_{t,t+\tau} c_{t+\tau} Markowitz_{t+\tau}\right]\| \\ &\leq \sum_{\tau} \|E_t[\tilde{N}_{t,t+\tau} q_{t+\tau}]\| \leq \sum_{\tau} E_t[\|\tilde{N}_{t,t+\tau} q_{t+\tau}\|^2]^{1/2} \end{aligned} \tag{C.46}$$

to prove the convergence of Q_t, we note that, since $\|q_{t+\tau}\|^2 \leq K$ for some $K > 0$, we have, by Lemma 6,

$$\begin{aligned} &\sum_{\tau} E_t[\|\tilde{N}_{t,t+\tau} q_{t+\tau}\|^2]^{1/2} = \sum_{\tau} E_t[q'_{t+\tau} \tilde{N}'_{t,t+\tau} \tilde{N}_{t,t+\tau} q_{t+\tau}]^{1/2} \\ &\leq \sum_{\tau} E_t[\|q_{t+\tau}\|^2 \operatorname{tr}(\tilde{N}'_{t,t+\tau} \tilde{N}_{t,t+\tau})]^{1/2} \leq K \sum_{\tau} E_t[\operatorname{tr}(\tilde{N}'_{t,t+\tau} \tilde{N}_{t,t+\tau})]^{1/2} \\ &= K \sum_{\tau} (\operatorname{tr} E_t[\tilde{N}'_{t,t+\tau} \tilde{N}_{t,t+\tau}])^{1/2} = K \sum_{\tau} (\operatorname{tr} E_t[\tilde{N}'_{t,t+\tau} \tilde{N}_{t,t+\tau}])^{1/2} . \end{aligned} \tag{C.47}$$

which follows from the made assumptions. Convergence of the series representation for π_t also follows from the made assumptions. The recursive equation implies

$$\pi_t = \sum_{\theta=0}^{\infty} N_{t-\theta,t} Q(s_{t-\theta}) . \tag{C.48}$$

Since Q_t is bounded by (C.18), we have $\|Q_t\| \leq K$ for some $K > 0$. By the same argument as above,

$$\begin{aligned} E[\|\pi_t\|] &= E[\|\left[\sum_{\theta=0}^{\infty} N_{t-\theta,t} Q(s_{t-\theta})\right]\|] \\ &\leq \sum_{\theta=0}^{\infty} E[\|N_{t-\theta,t} Q(s_{t-\theta})\|] \\ &\leq \sum_{\theta=0}^{\infty} E[\|N_{t-\theta,t} Q(s_{t-\theta})\|^2]^{1/2} \\ &\leq K \sum_{\theta=0}^{\infty} E[\|N_{t-\theta,t}\|^2]^{1/2} \\ &\leq K \sum_{\theta=0}^{\infty} (\operatorname{tr} E[N'_{t-\theta,t} N_{t-\theta,t}])^{1/2} . \end{aligned} \tag{C.49}$$

□

Recall that g_t is the diagonal matrix of $vec(g_t)$ on the diagonal. For the case when $\mu_t = \mu$ is constant, we have that

$$G_t = E_t[vac(g_{t+1}) vec(g_{t+1})'] = (1 + g_t^w)^{-2} (\Sigma + (1 + r_t^f + \mu_t)(1 + r_t^f + \mu_t)') . \tag{C.50}$$

When $g_t^w,\ r_t^f$, and μ_t are all constant, we get that $G_t = G$ is also constant, and we arrive at the following result, which is a direct consequence of Lemma 5 because $\bar{\Lambda}_t \ = \ E_t[g_{t+1}\Lambda g_{t+1}] = \Lambda \circ G$.

Lemma 7 *For simplicity, we normalize $\gamma/w = 1$. Suppose that $\mu_t = \mu$, g_t^w, r_t^f are constant. Let $\bar{\Lambda} \ = \ \Lambda \circ G$. For any solution m to*

$$m \ = \ (\Sigma + \Lambda + \bar{\Lambda})^{-1}\left(((\Lambda m) \circ G)m + \Lambda\right), \tag{C.51}$$

define

$$N_{t,t+\tau} \ = \ \prod_{\theta=0}^{\tau-1} m\, g_{t+\theta+1},\ N_{t,t} \ = \ I\,, \tag{C.52}$$

and

$$\tilde{N}_{t,t+\tau} \ = \ \prod_{\theta=0}^{\tau-1} m\,\Lambda^{-1} g_{t+\theta+1}\Lambda\ ,\ \tilde{N}_{t,t} \ = \ I\,. \tag{C.53}$$

Suppose that

$$\sum_{\tau=1}^{\infty} \|E[N'_{t,t+\tau}N_{t,t+\tau}]\|^{1/2} < \infty \tag{C.54}$$

and

$$esssup \sum_{\tau=1}^{\infty} \|E_t[\tilde{N}'_{t,t+\tau}\tilde{N}_{t,t+\tau}]\|^{1/2} < \infty\,. \tag{C.55}$$

Define

$$c \ = \ m\Lambda^{-1}\Sigma \tag{C.56}$$

and

$$Q \ = \ (I - m\Lambda^{-1}\bar{g}\Lambda)^{-1} c Markowitz,\ Markowitz \ = \ \Sigma^{-1}\mu\,, \tag{C.57}$$

and let

$$\pi_t \ = \ \pi(s_t, s_{t-}) \ = \ \sum_{\theta=0}^{\infty} N_{t-\theta,t} Q\,. \tag{C.58}$$

Then, all series converge in L_2, and π_t is optimal among all bounded stationary processes π_t. Furthermore, it satisfies the recursive relationship

$$\pi(s_t, s_{t-}) \ = \ Q \ + \ m g_t \pi(s_{t-1}, s_{t-1-})\,. \tag{C.59}$$

C.3 Properties the Discount Factor m

This section shows some useful properties of the discount factor m solving (C.51). We start with the following observation

Lemma 8 *For simplicity, we normalize $\gamma/w = 1$. A matrix-valued function $m(s_t) = m_t$ solves*

$$m_t \;=\; (\Sigma + \Lambda + \bar{\Lambda}_t)^{-1}\left(E_t[g_{t+1}\Lambda m_{t+1} g_{t+1}]m_t + \Lambda\right) \tag{C.60}$$

if and only if $\tilde{m}_t \;=\; \Lambda^{1/2} m_t \Lambda^{-1/2}$ solves

$$\tilde{m}_t \;=\; \left(\Lambda^{-1/2}\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}E_t[g_{t+1}\Lambda^{1/2}(I - \tilde{m}_{t+1})\Lambda^{1/2} g_{t+1}]\Lambda^{-1/2}\right)^{-1}. \tag{C.61}$$

In particular, if G_t is constant, then a matrix m solves

$$m \;=\; (\Sigma + \Lambda + \bar{\Lambda})^{-1}\left(((\Lambda m) \circ G)m + \Lambda\right) \tag{C.62}$$

with $\bar{\Lambda} = \Lambda \circ G$ if and only if the matrix $\tilde{m} = \Lambda^{1/2} m \Lambda^{-1/2}$ solves

$$\tilde{m} \;=\; \left(\Lambda^{-1/2}\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}(G \circ (\Lambda^{1/2}(I - \tilde{m})\Lambda^{1/2}))\Lambda^{-1/2}\right)^{-1}. \tag{C.63}$$

Proof of Lemma 8. We have

$$(\Sigma + \Lambda + \bar{\Lambda}_t)m_t \;=\; \left(E_t[g_{t+1}\Lambda m_{t+1} g_{t+1}]m_t + \Lambda\right). \tag{C.64}$$

Writing $m_t \;=\; \Lambda^{-1/2}\tilde{m}_t\Lambda^{1/2}$, we get

$$(\Sigma + \Lambda + \bar{\Lambda}_t)\Lambda^{-1/2}\tilde{m}_t\Lambda^{1/2} \;=\; \left(E_t[g_{t+1}\Lambda\Lambda^{-1/2}\tilde{m}_{t+1}\Lambda^{1/2} g_{t+1}]\Lambda^{-1/2}\tilde{m}_t\Lambda^{1/2} + \Lambda\right). \tag{C.65}$$

Multiplying by $\Lambda^{-1/2}\tilde{m}_t^{-1}$ from the right and by $\Lambda^{-1/2}$ from the left, we get

$$\Lambda^{-1/2}(\Sigma + \Lambda + \bar{\Lambda}_t)\Lambda^{-1/2} \;=\; \left(\Lambda^{-1/2}E_t[g_{t+1}\Lambda^{1/2}\tilde{m}_{t+1}\Lambda^{1/2} g_{t+1}]\Lambda^{-1/2} + \tilde{m}_t^{-1}\right). \tag{C.66}$$

This is equivalent to

$$\tilde{m}_t^{-1} \;=\; \Lambda^{-1/2}(\Sigma + \Lambda + \bar{\Lambda}_t - E_t[g_{t+1}\Lambda^{1/2}\tilde{m}_{t+1}\Lambda^{1/2} g_{t+1}])\Lambda^{-1/2}, \tag{C.67}$$

which is, in turn, equivalent to

$$\tilde{m}_t \;=\; \left(\Lambda^{-1/2}\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}E_t[g_{t+1}\Lambda^{1/2}(I - \tilde{m}_{t+1})\Lambda^{1/2} g_{t+1}]\Lambda^{-1/2}\right)^{-1}. \tag{C.68}$$

In the case when $m_t = m$ is constant and $G_t = G$ is constant, we get

$$E_t[g_{t+1}\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2}g_{t+1}] \;=\; G\circ(\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2})\,,$$

and we get the required. □

Recall that $\mathcal{S}(0,1)$ is the set of symmetric, positive semi-definite matrices with eigenvalues in $(0,1)$. We will also need the following lemma.

Lemma 9 *We have*

$$E[\|A\|^2] \;\le\; \operatorname{tr} E[A'A] \;\le\; n\|E[A'A]\| \;\le\; nE[\|A\|^2]$$

for any random matrix A where n is the dimension of $A'A$.

Proposition 9 *Suppose that there exists a solution $\tilde{m}\in\mathcal{S}(0,1)$ to (C.63). Let $q_*<1$ be the largest eigenvalue of $\tilde{m}$. Let $m=\Lambda^{-1/2}\tilde{m}\Lambda^{1/2}$ and*

$$\mathcal{Q}_{t-\theta,t} \;=\; \left(\prod_{\tau=1}^{\theta} m\, g_{t-\tau+1}\right). \tag{C.69}$$

Then,

$$\lim_{\theta\to\infty} q_*^{-\theta}E[\|\mathcal{Q}_{t-\theta,t}(\nu)\|^2] \;=\; 0\,. \tag{C.70}$$

Similarly, if we define

$$\hat{\mathcal{Q}}_{t,t+\theta} \;=\; \prod_{\tau=1}^{\theta}(m\Lambda^{-1}g_{t+\tau-1}\Lambda)\,.$$

Then,

$$\lim_{\theta\to\infty} q_*^{-\theta}E[\|\hat{\mathcal{Q}}_{t,t+\theta}\|^2] \;=\; 0\,. \tag{C.71}$$

Proof of Proposition 9. For simplicity, we normalize $\gamma/w=1$. Recall that $\mathcal{S}$ is the set of symmetric matrices, and $\mathcal{S}(a,b)$ is the set of positive semi-definite matrices with eigenvalues between a and b.

Equation (C.63) can be rewritten as

$$\tilde{m}\Big(\Lambda^{-1/2}\Sigma\Lambda^{-1/2}+I+\Lambda^{-1/2}E_t[g_{t+1}\Lambda^{1/2}(I-\tilde{m}_{t+1})\Lambda^{1/2}g_{t+1}]\Lambda^{-1/2}\Big)\tilde{m} \;=\; \tilde{m}\,. \tag{C.72}$$

Define the map

$$\Xi(Z) \;=\; \tilde{m}\Lambda^{-1/2}E_t[g_{t+1}\Lambda^{1/2}Z\Lambda^{1/2}g_{t+1}]\Lambda^{-1/2}\tilde{m}, \tag{C.73}$$

mapping the cone of positive semi-definite matrices into itself. Then, (C.72) implies that

$$\Xi(I-\tilde{m}) \;=\; \tilde{m}-\tilde{m}^2-\tilde{m}\Lambda^{-1/2}\Sigma\Lambda^{-1/2}\tilde{m} \;<\; \tilde{m}(I-\tilde{m}) \;\le\; q_*(I-\tilde{m})\,. \tag{C.74}$$

Now, the map Ξ leaves the proper cone $\mathcal{S}(0,+\infty)$ invariant, and hence, by the Krein and Rutman (1950) theorem, its spectral radius corresponds to a strictly positive eigenvalue $\lambda_* > 0$. Let $Z \in \mathcal{S}(0,+\infty)$ be the corresponding eigenvector. Then,

$$\Xi(Z) \;=\; \lambda_* \, Z \,.$$

Since $I - \tilde{m}$ is strictly positive definite, there exists a constant $a_* > 0$ such that $aZ \leq I - \tilde{m}$ if and only if $a \leq a_*$. Then,

$$\lambda a_* Z = \Xi(a_* Z) \;\leq\; \Xi(I - \tilde{m}) \;<\; q_* \, (I - \tilde{m})$$

Thus, $(\lambda a_*/q_*)Z \;\leq\; I - \tilde{m}$ implying that, by the definition of a_*, we must have $\lambda/q_* < 1$.

Note also that the transformation

$$\hat{\Xi}(Z) \;=\; A^{-1}\tilde{m}E_t[\Lambda^{-1/2}g_{t+1}\Lambda^{1/2}AZA'\Lambda^{1/2}g_{t+1}\Lambda^{-1/2}]\tilde{m}(A')^{-1}$$

is similar to Ξ for any invertible matrix A. Hence, Ξ and $\hat{\Xi}$ have the same spectral radius. Pick $A = \tilde{m}\Lambda^{-1/2}$. Then,

$$\hat{\Xi}(Z) \;=\; E_t[g_{t+1}\Lambda^{1/2}\tilde{m}\Lambda^{-1/2}Z\Lambda^{-1/2}\tilde{m}\Lambda^{1/2}g_{t+1}] \;=\; E_t[g_{t+1}m'Zmg_{t+1}] \,.$$

By direct calculation,

$$E[\mathcal{Q}'_{t-\theta,t}\mathcal{Q}_{t-\theta,t}] \;=\; \hat{\Xi}^{\theta}(I) \,,$$

and

$$E[\hat{\mathcal{Q}}_{t,t+\theta}\hat{\mathcal{Q}}'_{t,t+\theta}] \;=\; \Xi^{\theta}(I) \,,$$

and the claim follows from Lemma 9. □

C.4 Proofs of Propositions 2 and 3

Lemma 10 *For simplicity, we normalize $\gamma/w = 1$. Consider the map F mapping the convex set of $\mathcal{S}(0,1)$-valued matrix functions into itself and defined via*

$$F(\tilde{m}_t) \;=\; \left(\Lambda^{-1/2}\Sigma\Lambda^{-1/2} + I + \Lambda^{-1/2}E_t[g_{t+1}\Lambda^{1/2}(I - \tilde{m}_{t+1})\Lambda^{1/2}g_{t+1}]\Lambda^{-1/2}\right)^{-1} . \qquad \text{(C.75)}$$

This map is strictly monotone increasing in the positive semi-definite order and hence has at least one fixed point in $\mathcal{S}(0,1)$. The set of fixed points has a unique maximal and a unique minimal element. The minimal element is obtained by iterating F on 0. The maximal element is obtained by iterating F on I.

Proof of Lemma 10. The proof follows directly from the fact that the map $A \to A^{-1}$ is monotone decreasing in the positive semi-definite order, and the same is true for the map $\tilde{m}_{t+1} \to E_t[g_{t+1}\Lambda^{1/2}(I - \tilde{m}_{t+1})\Lambda^{1/2}g_{t+1}]$. □

When μ is stochastic, things are a bit more tricky, as shown by the following lemma.

Lemma 11 *Suppose that $\mu(s_t) \;=\; \epsilon\tilde{\mu}(s_t)$, $g_t^w = g^w + O(\varepsilon)$, $r_t^f = r^f + O(\varepsilon)$. Then,*

$$G_t \;=\; G \;+\; O(\epsilon) \qquad \text{(C.76)}$$

where $G = E[vec(g_t)vec(g_t)']$ *and hence, for every solution* $\tilde{m} \in \mathcal{S}(0,1)$ *to* (C.63) *and any sufficiently small* $\epsilon > 0$ *there exists a unique solution* $\tilde{m}_t$ *to* (C.61) *satisfying*

$$\tilde{m}_t \;=\; \tilde{m} \;+\; O(\varepsilon)\,. \tag{C.77}$$

Proof of Lemma 11. The proof follows directly from the implicit function theorem and the fact that the map F from Lemma 10 is strictly monotone on $\mathcal{S}(0,1)$ and (by direct calculation) has a non-degenerate Jacobian. □

Proof of Proposition 2. By Lemma 10, there exists a $\tilde{m} \in \mathcal{S}(0,1)$ solving (C.63). By Proposition 9, the technical conditions (C.54) and (C.55) are satisfied. Lemma 8 implies that m solves (C.51) and hence Lemma 7 implies that the policy π is optimal. Its uniqueness follows from the strict concavity of the objective: If there were two solutions, both would be optimal, which is impossible for a strictly concave objective. □

Proof of Lemma 1. Uniqueness of the solution to (14) follows from the proof of Proposition 2: Indeed, if there are two solutions to (14) then, by the proof of Proposition 2, each of them corresponds to a different optimal policy which is impossible by strict concavity. It remains to prove that all eigenvalues of $\Lambda^{-1/2}\tilde{m}\Lambda^{-1/2}\bar{g}\Lambda$ are between zero and one because then, using that Λ is diagonal, we get $m\Lambda^{-1}\bar{g}\Lambda = \Lambda^{-1/2}\tilde{m}\Lambda^{1/2}\Lambda^{-1}\bar{g}\Lambda = \Lambda^{-1/2}\tilde{m}\Lambda^{-1/2}\bar{g}\Lambda$. To prove the desired claim, we note that, by Proposition 9, we have

$$\hat{\mathcal{Q}}_{t,t+\theta} \;=\; \prod_{\tau=1}^{\theta}(m\Lambda^{-1}g_{t+\tau-1}\Lambda)$$

satisfies

$$\lim_{\theta\to\infty} q_*^{-\theta}E[\|\hat{\mathcal{Q}}_{t,t+\theta}\|^2] \;=\; 0\,. \tag{C.78}$$

for some $q_* \in (0,1)$. Thus, for any vector v, we have

$$\|E[\hat{\mathcal{Q}}_{t,t+\theta}]v\| \;=\; \|E[\hat{\mathcal{Q}}_{t,t+\theta}v]\| \leq \;\|v\|E[\|\hat{\mathcal{Q}}_{t,t+\theta}\|] \;\leq\; \|v\|E[\|\hat{\mathcal{Q}}_{t,t+\theta}\|^2]^{1/2} \tag{C.79}$$

By assumption, $E_t[g_{t+1}] = \bar{g}$ is non-stochastic and, hence, using the law of iterated expectations, we get

$$E[\hat{\mathcal{Q}}_{t,t+\theta}] \;=\; \prod_{\tau=1}^{\theta}(m\Lambda^{-1}\bar{g}\Lambda) \;=\; (m\Lambda^{-1}\bar{g}\Lambda)^{\theta}.$$

Let v be the eigenvector for the largest eigenvalue of $m\Lambda^{-1}\bar{g}\Lambda$ and let λ be this eigenvalue. Then,

$$\lambda^{\theta}q_*^{-\theta} \;\to\; 0$$

as $\theta \to \infty$ implies $\lambda < q_* < 1$. The proof is complete. □

Proof of Proposition 3. By Lemma 10, there exists a $\tilde{m} \in \mathcal{S}(0,1)$ solving (C.63). By Lemma 11, there exists a $\tilde{m}_t$ solving (C.61) satisfying $m_t = \tilde{m} \;+\; O(\varepsilon)$. By a small modification of the proof of Lemma 9, the technical conditions (C.18) and (C.19) are satisfied.

Lemma 8 implies that m_t solves (C.60) and hence Lemma 5 implies that the policy

$$\pi_t^* \;=\; \pi(s_t, s_{t-}) \;=\; \sum_{\theta=0}^{\infty} N_{t-\theta,t} Q(s_{t-\theta})\,. \tag{C.80}$$

is optimal with

$$Q_t \;=\; E_t\left[\sum_{\tau=0}^{\infty} \tilde{N}_{t,t+\tau} c_{t+\tau} Markowitz_{t+\tau}\right] \tag{C.81}$$

and

$$N_{t,t+\tau} \;=\; \prod_{\theta=0}^{\tau-1} m_{t+\theta}\, g_{t+\theta+1},\;\; N_{t,t} \;=\; I\,, \tag{C.82}$$

and

$$\tilde{N}_{t,t+\tau} \;=\; \prod_{\theta=0}^{\tau-1} m_{t+\theta}\, \Lambda^{-1} g_{t+\theta+1} \Lambda\;,\;\; \tilde{N}_{t,t} \;=\; I\,, \tag{C.83}$$

and $c_t = m_t \Lambda^{-1} \Sigma$. Its uniqueness follows from the strict concavity of the objective. Now, substituting $m_t = m + O(\varepsilon)$ into these equations, we get that π_t^* differs from (20) by $O(\varepsilon)$ (technical conditions (C.18)-(C.19) ensure that the infinite sum also is $O(\varepsilon)$.) The proof is complete. □

C.5 Proofs of Propositions 6 and 7

We have $m = \Lambda^{-1/2} \tilde{m} \Lambda^{1/2}$ and hence m and $\tilde{m}$ have identical eigenvalues, and the claim of Proposition 6 follows from Lemma 4 and the monotonicity of the map F.

The convergence of $m = \Lambda^{-1/2} \tilde{m} \Lambda^{1/2}$ to zero when $w \to 0$ follows directly from (C.63) rewritten for the general case when $w/\gamma \neq 1$:

$$\tilde{m} \;=\; \left(\Lambda^{-1/2}\Sigma\Lambda^{-1/2}(\gamma/w)+I+\Lambda^{-1/2}(G\circ(\Lambda^{1/2}(I-\tilde{m})\Lambda^{1/2}))\Lambda^{-1/2}\right)^{-1} \;\leq\; w\gamma^{-1}\Lambda^{1/2}\Sigma^{-1}\Lambda^{1/2}\,. \tag{C.84}$$

When $w \to \infty$, this equation takes the form

$$\tilde{m} \;=\; \left(I + \Lambda^{-1/2}(G \circ (\Lambda^{1/2}(I - \tilde{m})\Lambda^{1/2}))\Lambda^{-1/2}\right)^{-1} \tag{C.85}$$

One solution to this equation is $\tilde{m} = I$, but it would lead to explosive behavior. It turns out that, strikingly, $\tilde{m}$, in fact, converges to a different solution.

To prove the desired convergence, we need to show that the technical conditions (C.54) and (C.55) hold uniformly when $w \to \infty$. By Proposition 9, to this end we need to show that q_*, the maximal eigenvalue of $\tilde{m}$ stays uniformly bounded away from 1. This follows from Lemma 4: Since $\tilde{m} \leq \tilde{m}(\bar{g})$, it suffices to establish this fact for $\tilde{m}(\bar{g})$. Let $\xi = \bar{g}$. From

the proof of Lemma 3, we have

$$\tilde{m}(\xi) \;=\; \xi^{-1/2}\hat{m}\xi^{-1/2}$$

where

$$\hat{m} \;=\; 0.5(\hat{\Sigma} - (\hat{\Sigma}^2 - 4I)^{1/2}) \;=\; 2(\hat{\Sigma} + (\hat{\Sigma}^2 - 4I)^{1/2})^{-1} \tag{C.86}$$

and

$$\hat{\Sigma} \;=\; \xi^{-1/2}\tilde{\Sigma}\xi^{-1/2} \;=\; \gamma w^{-1}\xi^{-1/2}\Lambda^{-1/2}\Sigma\Lambda^{-1/2}\xi^{-1/2} + \xi^{-1} + \xi\,.$$

and hence $\hat{\Sigma} \to \xi^{-1} + \xi$ when $w \to \infty$. Thus,

$$\tilde{m} \;\to\; 0.5\xi^{-1}(\xi^{-1} + \xi - |\xi - \xi^{-1}|)$$

Under our assumption $\bar{g} = \xi > 1$. Then, we get $\tilde{m} = \xi^{-2} < 1$. That is, $\tilde{m}$ stays uniformly bounded away from I. Then,

$$wc \;=\; wm(\gamma/w)\Lambda^{-1}\Sigma \;\to\; m\gamma\Lambda^{-1}\Sigma \tag{C.87}$$

and, hence,

$$wQ \;=\; (I - m\Lambda^{-1}\bar{g}\Lambda)^{-1}wcMarkowitz \tag{C.88}$$

converges to a finite limit, and so does

$$w\pi_t \;=\; \sum_{\theta=0}^{\infty} N_{t-\theta,t}wQ\,. \tag{C.89}$$

C.6 On the Optimality of Portfolio-ML

Given a function $A(\cdot)$, denote

$$\Pi_t^*(A(\cdot),\ \vec{s}) \;=\; \sum_{\theta=0}^{\infty}\left(\prod_{\tau=1}^{\theta} m\, g_{t-\tau+1}\right)(I-m)A(s_{t-\theta})\,. \tag{C.90}$$

We formalize the above observations in the following lemma.

Lemma 12 *Suppose that $\mu(s_t) = \varepsilon\tilde{\mu}(s_t)$ where ε is a small number. Then, the solution to the aim optimization problem*

$$\begin{aligned}
&\max_{A(\cdot)} E\Bigg[\mu(s_t)'\Pi_t^*(A(\cdot),\ \vec{s}) \\
&-\frac{w}{2}\Big(\Pi_t^*(A(\cdot),\ \vec{s}_t) - g_t\Pi_t^*(A(\cdot),\ \vec{s}_{t-1})\Big)'\Lambda\Big(\Pi_t^*(A(\cdot),\ \vec{s}_t) - g_t\Pi_t^*(A(\cdot),\ \vec{s}_{t-1})\Big) \\
&-\frac{\gamma}{2}\left(\Pi_t^*(A(\cdot),\ \vec{s}_t)\right)'\Sigma\left(\Pi_t^*(A(\cdot),\ \vec{s}_t)\right)\Bigg]
\end{aligned} \tag{C.91}$$

coincides with (21) *up to terms of the order* ε^2.

Lemma 12 reduces the infeasible (due to infinite history dependence) portfolio optimization problem (9) to a feasible aim optimization problem (C.91), where only the function $A(s_t)$ of the current state needs to be optimized. As we now show, it is possible to use machine learning methods to further reduce (C.91) to a linear-quadratic problem that can be solved analytically. We will need the following assumptions.

Assumption 1 *Let* s_{-i} *denote the vector of signals for all stocks except* i. *There exists a function* $a(s_i, s_{-i})$ *such that* $(A(s_t))_i = a(s_{i,t},\, s_{-i,t})$.

Assumption 1 is not restrictive and naturally holds whenever $\Lambda_{i,j} = \Lambda(s_{i,t}, s_{j,t})$ and $\Sigma_{i,j} = \Sigma(s_{i,t}, s_{j,t})$ for some universal functions Λ, Σ. It ensures that the dependence of the aim on signals is the same for all stocks.

Assumption 2 *Suppose that a family of functions* $\{f_k(s)\}_{k\geq 1}$ *has the universal approximation property: For any* $\epsilon > 0$ *there exists a* $P > 0$ *and a vector* $\beta \in \mathbb{R}^P$ *such that*

$$\|a(s) - \sum_k \beta_k f_k(s)\|_2 \;\leq\; \epsilon\,. \tag{C.92}$$

Let now $F(s) = (f_k(s_i, s_{-i}))_{i,k=1}^{n,P}$. Then, Assumptions 1 and 2 imply the existence of a vector β such that

$$\|A(s) - F(s)\beta\|_2 \;\leq\; \epsilon\,, \tag{C.93}$$

and hence we can rewrite (C.90) as

$$\pi_t \;=\; X_t\beta \;+\; O(\epsilon)\,, \tag{C.94}$$

where

$$X_t \;\equiv\; \left[\sum_{\theta=0}^{\infty}\left(\prod_{\tau=1}^{\theta} m\, g_{t-\tau+1}\right)(I-m)f(s_{t-\theta})\right]\,. \tag{C.95}$$

Using this formulation for π_t in the objective (C.91), we can rewrite it as

$$\begin{aligned}
&E\left[r'_{t+1}X_t\beta - \frac{\gamma}{2}\beta' X_t'\Sigma X_t\beta - \frac{w}{2}\left(X_t\beta - g_t X_{t-1}\beta\right)'\Lambda\left(X_t\beta - g_t X_{t-1}\beta\right)\right] \;+\; \lambda\|\beta\|^2 \\
&= E\left[\underbrace{r'_{t+1}X_t}_{\equiv\tilde{r}'_{t+1}}\beta - \frac{1}{2}\beta'\underbrace{\left[\gamma X_t'\Sigma X_t + w(X_t - g_tX_{t-1})'\Lambda(X_t - g_tX_{t-1})\right]}_{\equiv\tilde{\Sigma}_t}\beta\right] \\
&\equiv E[\tilde{r}'_{t+1}]\beta - \frac{1}{2}\beta' E[\tilde{\Sigma}_t]\beta \;+\; \lambda\|\beta\|^2
\end{aligned} \tag{C.96}$$

and the optimal β is given by

$$\beta^* \;=\; (\lambda I + E[\tilde{\Sigma}_t])^{-1}E[\tilde{r}_{t+1}]\,. \tag{C.97}$$

Uniform boundedness of all coefficients implies that the solution to the approximate optimization problem achieves approximate optimum in (C.91). Thus, we can maximize utility by maximizing this quadratic equation in the unknown parameter vector β.

Proposition 10 (Portfolio-ML) *Suppose that $\mu(s_t) = \varepsilon\tilde{\mu}(s_t)$, where ε is a small number. Let β_T be a finite sample counterpart of (C.97). Then, in the limit as $T \to \infty$, $\lambda \to 0$ β_T converges to β^*. Furthermore, the optimal portfolio (24) achieves the optimal utility (9) up to an error of the order $\varepsilon^2 + \epsilon$, where ϵ is defined in (C.92).*

Proof of Proposition 4. The proof follows directly from Proposition 10.

□

C.7 Random Features and their Universal Approximation Properties

Suppose that the Markov state satisfies $s \in \mathbb{R}^p$ for some $p \in \mathbb{N}$. Let $\nu_*(s)$ be the stationary distribution of s.

Let $p(\theta)d\theta,\ \theta \in \mathbb{R}^d$ be a distribution. A random feature model is a family of functions $\sigma(s,\theta):\ \mathbb{R}^p \times \mathbb{R}^d \to \mathbb{R}$. It defines positive-definite kernel $K(s_1,s_2):\ \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$

$$K(s_1,s_2) \ = \ \int_{\mathbb{R}^d} \sigma(s_1,\theta)\sigma(s_2,\theta)p(\theta)d\theta\,. \tag{C.98}$$

For simplicity, we assume that σ is uniformly bounded (this is the case for the random Fourier features $\cos(s'\theta),\ \sin(s'\theta)$ that we use in our empirical exercise). Then,

$$\int\int K(s_1,s_2)^2 d\nu_*(s_1)d\nu_*(s_2) \ < \ \infty \tag{C.99}$$

and therefore, by the Mercer theorem (Mohri et al. (2018)), there exists a basis of eigenfunctions $\psi_j(s),\ j \geq 0$, such that

$$\int K(s_1,s_2)\psi_j(s_2)d\nu_*(s_2) \ = \ \lambda_j\psi_j(s_1)\,, \tag{C.100}$$

and

$$K(s_1,s_2) \ = \ \sum_{j=0}^{\infty} \lambda_j\psi_j(s_1)\psi_j(s_2)\,, \tag{C.101}$$

where $\lambda_j > 0$ are the eigenvalues of K, satisfying

$$\sum_{j=0}^{\infty} \lambda_j \ = \ \int\int K(s_1,s_2)d\nu_*(s_1)d\nu_*(s_2) \ < \ \infty\,. \tag{C.102}$$

Given the kernel, one can define the corresponding *Reproducing Kernel Hilbert Space*

$$\mathcal{H} \ = \ \left\{ f:\ \|f\|_{\mathcal{H}}^2 \ = \ \sum_{j=0}^{\infty} \lambda_j^{-1}\langle f,\psi_j\rangle_{L_2}^2 \ < \ \infty \right\}, \tag{C.103}$$

where

$$\langle f,\psi_j\rangle_{L_2} \ = \ \int f(s)\psi_j(s)d\nu_*(s)\,. \tag{C.104}$$

It is known that the following is true (see, e.g., Maddalena et al. (2021), Liu et al. (2021))

Proposition 11 *Suppose that the data satisfies $y_i = f(s_i) + \varepsilon_i$ where samples $i = 1, \cdots, N$ are i.i.d., the noise is uniformly bounded in L_2, and and f belongs to the Reproducing Kernel Hilbert Space $\mathcal{H}$. Then, the kernel ridge regression estimator*

$$\hat{f}(s) \;=\; K(s,S)(K(S,S)+zI)^{-1}y \tag{C.105}$$

converges to $f(s)$ in L_2 as $N \to \infty,\ z \to 0$, where $K(S,S) = (K(s_i,s_j))_{i,j=1}^N$ and $K(s,S) = (K(s,s_j))_{j=1}^N$.

Thus, in large enough samples, kernel ridge regression recovers the truth. As discovered by Rahimi and Recht (2007), there is a tight connection between kernel ridge regression and simple, linear regression with random features. It is based on the Monte-Carlo approximation of the kernel:

$$K(s_1,s_2) \;=\; \int_{\mathbb{R}^d} \sigma(s_1,\theta)\sigma(s_2,\theta)p(\theta)d\theta \;\approx\; \frac{1}{P}\sum_{k=1}^{P}\sigma(s_1,\theta_k)\sigma(s_2,\theta_k) \;=\; \hat{K}(s_1,s_2;P) \tag{C.106}$$

In this case, defining the feature vectors

$$X_i \;=\; X(s_i) \;=\; (\sigma(s_i,\theta_k))_{k=1}^P \;\in \mathbb{R}^P\,, \tag{C.107}$$

we get that the kernel ridge regression using the approximate kernel (C.106) is equivalent to a simple, linear ridge regression with random features, as formalized by the following result (see Rahimi and Recht (2007)).

Lemma 13 *Consider a linear regression using X :*

$$\hat{f}_P(s) \;=\; \hat{\beta}'X(s), \tag{C.108}$$

where

$$\hat{\beta} \;=\; (zI + X(S)X(S)')^{-1}X(S)y,\ \ X(S) \;=\; (\sigma(s_i,\theta_k))_{k,i=1}^{P,N} \tag{C.109}$$

Then, in the limit as $P \to \infty$, we have $\hat{f}_P(s) \to \hat{f}(s)$, when θ_i are sampled independently from $p(\theta)$.

Thus, with sufficiently many random features, the linear regression with these features converges to the outcome of the kernel ridge regression. However, the exact nature of this convergence, as well as the dependence on the sample size N and the dimension d of the data, is complex, as is shown by Mei et al. (2022).

To understand the convergence in practice, we show a simple one-dimensional example. Specifically, we define the function $f(x) = 10x + 0.5x^2 - 0.01x^3 - 1.08^x + 100\sin(x)$, and sample 1000 observation from this function by adding normally distributed mean-zero noise with a variance of 150^2. We then fit a ridge regression with ridge penalty of $z = 1$, to $2^1, 2^2, \ldots, 2^6$ random features generated using the function from (39) with $\eta = 1$. Figure C.1 shows that the ridge regression converges to the true function with 64 random features.

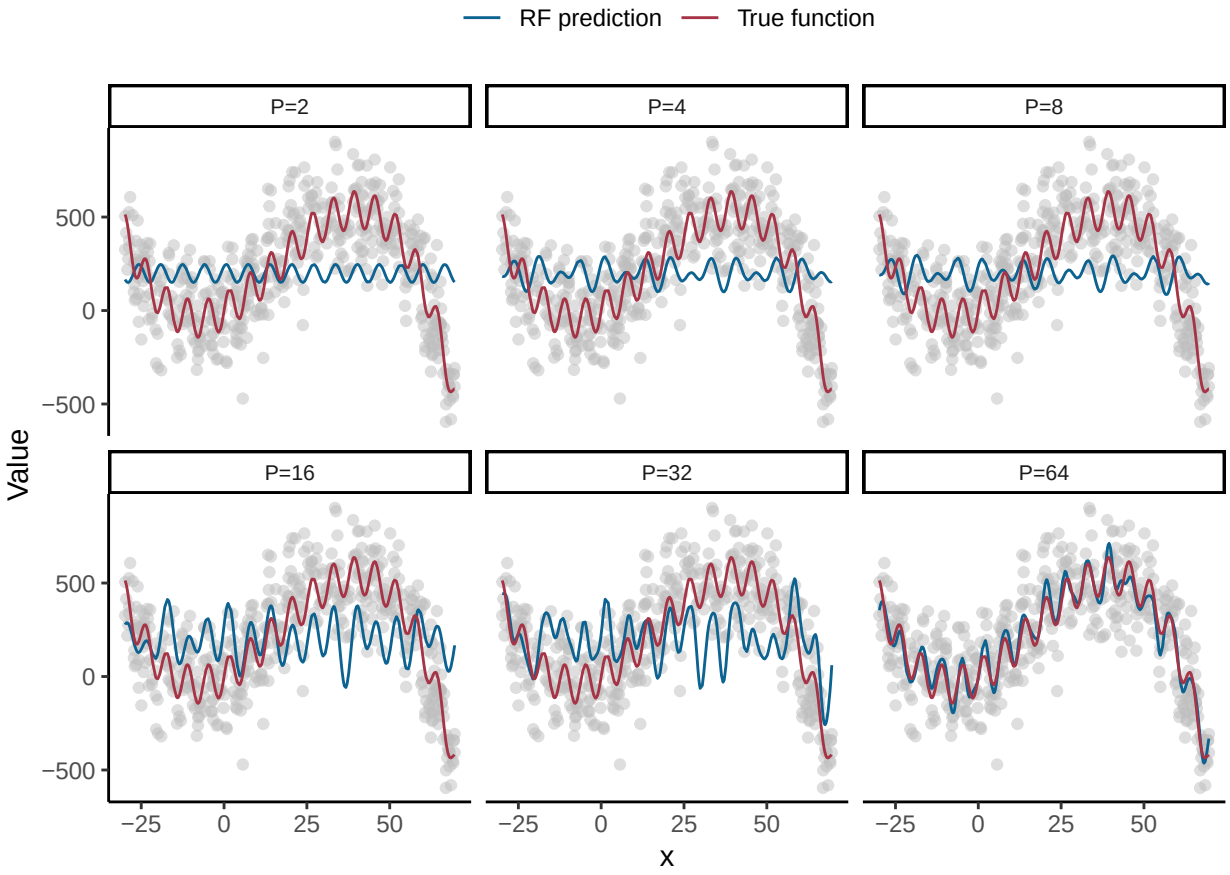

Figure C.1: Random feature regression example

Note: The figure shows a ridge regression fitted to a varying number of random features. The red line shows the true data generating function, defined as $f(x) = 10x + 0.5x^2 - 0.01x^3 - 1.08^x + 100\sin(x)$. The grey dots show 1,000 samples from this function, generated by adding normally distributed mean-zero noise with a variance of 150^2. The blue line shows the prediction from a ridge regression fitted on $P = 2^1, 2^2, \ldots, 2^6$ random features, respectively, as indicated by the panel title.

C.8 Proofs related to Economic Feature Importance

The optimal portfolio admits a simpler analytical expression when transaction costs are small, as seen in the following result.

Proposition 12 *Suppose that $\|\Lambda\|$ is small and Λ is diagonal. Then, the aim portfolio A_t is given by*

$$\begin{aligned} A_t &= Markowitz_t \\ &+ \Sigma^{-1} E_t[\Lambda g_{t+1} Markowitz_{t+1} - g_{t+1}\Lambda g_{t+1} Markowitz_t)] + O(\|\Lambda\|^2)\,, \end{aligned} \tag{C.110}$$

while the investor's utility is given by the same expression as in Proposition 5.

Proof of Proposition 5 and Proposition 12. Let $\varepsilon = \|\Lambda\|$. Then, $\|m_t\| = O(\varepsilon)$ and we

have

$$m_t \;=\; \Sigma^{-1}\Lambda - \Sigma^{-1}(\Lambda + E_t[g_{t+1}\Lambda g_{t+1}])\Sigma^{-1}\Lambda \;+\; O(\varepsilon^3)$$

and

$$(I - m_t)^{-1} \;=\; I + m_t \;+\; O(\varepsilon^2) \;=\; I \;+\; \Sigma^{-1}\Lambda \;+\; O(\varepsilon^2),$$

and

$$c_t = m_t\Lambda^{-1}\Sigma_t. \;=\; I - \Sigma^{-1}(\Lambda + E_t[g_{t+1}\Lambda g_{t+1}]) \;+\; O(\varepsilon^2)\,.$$

Let $\nu_t = Markowitz_t$. Then,

$$\begin{aligned}
A_t \;&=\; (I - m_t)^{-1}\sum_{\tau=0}^{\infty} E_t\left[M_{t,t+\tau}\, c_{t+\tau}\, Markowitz_{t+\tau}\right] \\
&=\; (I - m_t)^{-1}(c_t\nu_t + E_t[m_t\Lambda^{-1}\Lambda g_{t+1}c_{t+1}\nu_{t+1}]) \;+\; O(\varepsilon^2) \\
&=\; (I + \Sigma^{-1}\Lambda)\Big(c_t\nu_t \;+\; E_t[\Sigma^{-1}\Lambda\Lambda^{-1}\Lambda g_{t+1}c_{t+1}\nu_{t+1}]\Big) \;+\; O(\varepsilon^2) \\
&=\; (I + \Sigma^{-1}\Lambda)\Big(\Big(I - \Sigma^{-1}(\Lambda + E_t[g_{t+1}\Lambda g_{t+1}])\Big)\nu_t \\
&+\; E_t[\Sigma^{-1}\Lambda\Lambda^{-1}\Lambda g_{t+1}\Big(I - \Sigma^{-1}(\Lambda + E_t[g_{t+1}\Lambda g_{t+1}])\;\Big)\nu_{t+1}]\Big) \;+\; O(\varepsilon^2) \\
&=\; (I - \Sigma^{-1}E_t[g_{t+1}\Lambda g_{t+1}])\nu_t \;+\; E_t[\Sigma^{-1}\Lambda g_{t+1}\nu_{t+1}] + \;O(\varepsilon^2) \\
&=\; \nu_t \;+\; \Sigma^{-1}E_t[\Lambda g_{t+1}\nu_{t+1} - g_{t+1}\Lambda g_{t+1}\nu_t)] \;+\; O(\varepsilon^2)\,.
\end{aligned} \tag{C.111}$$

and

$$\begin{aligned}
\pi_t \;&=\; m_t g_t\pi_{t-1} \;+\; (I - m_t)A_t \\
&=\; m_t g_t\nu_{t-1} + (I - m_t)(\nu_t \;+\; \Sigma^{-1}E_t[\Lambda g_{t+1}(\nu_{t+1} - g_{t+1}\nu_t)]) \;+\; O(\varepsilon^2) \\
&=\; \nu_t \;+\; \Sigma^{-1}E_t[\Lambda g_{t+1}(\nu_{t+1} - g_{t+1}\nu_t)] + \Sigma^{-1}\Lambda(g_t\nu_{t-1} - \nu_t) \;+\; O(\varepsilon^2) \\
&=\; \nu_t \;+\; \xi_t \;+\; O(\varepsilon^2)\,.
\end{aligned} \tag{C.112}$$

Hence

$$\begin{aligned}
&E[\mu(x_t)'\pi_t - 0.5\pi_t'\Sigma\pi_t - 0.5(\pi_t - g_t\,\pi_{t-1})'\Lambda(\pi_t - \pi_{t-1})] \\
&=\; E[\mu(x_t)'\pi_t - 0.5\pi_t'(\Sigma + \Lambda + E_t[g_{t+1}\Lambda g_{t+1}])\pi_t + \pi_{t-1}'g_t\Lambda\pi_t] \\
&=\; E[\mu_t'(\nu_t + \xi_t) \\
&- 0.5(\nu_t + \xi_t)'(\Sigma + \Lambda + E_t[g_{t+1}\Lambda g_{t+1}])(\nu_t + \xi_t) \\
&+ (\nu_{t-1} + \xi_{t-1})'g_t\Lambda(\nu_t + \xi_t)] \\
&=\; 0.5E[\mu_t'\Sigma^{-1}\mu_t] \;+\; E[\mu_t'(\Sigma^{-1}E_t[\Lambda g_{t+1}(\nu_{t+1} - g_{t+1}\nu_t)] + \Sigma^{-1}\Lambda(g_t\nu_{t-1} - \nu_t))] \\
&- E[\nu_t'\Sigma\xi_t] \;-\; 0.5E[\nu_t'(\Lambda + E_t[g_{t+1}\Lambda g_{t+1}]))\nu_t] \\
&+ E[\nu_{t-1}'g_t\Lambda\nu_{t-1}] \\
&=\; 0.5E[\mu_t'\Sigma^{-1}\mu_t] \;-\; 0.5E[\nu_t'(\Lambda + E_t[g_{t+1}\Lambda g_{t+1}]))\nu_t] \;+ E[\nu_t'g_{t+1}\Lambda\nu_{t+1}] \\
&=\; 0.5E[\mu_t'\Sigma^{-1}\mu_t] \;- 0.5E[(\nu_t - g_t\nu_{t-1})'\Lambda(\nu_t - g_t\nu_{t-1})] + O(\varepsilon^2)\,.
\end{aligned}$$

□

D Data and Additional Empirical Results

D.1 Information of Stock Characteristics (Features)

Table D.1 shows the security characteristics that we use as features for all portfolio methods. The features are a subset of the 153 characteristics used in Jensen et al. (2023) plus 1-year trailing volatility (rvol_252d),[25] where the subset is chosen to have sufficient coverage in the early parts of our sample. Specifically, we select all features with a non-missing value for at least 70% of our sample by the end of 1952. The theme assignments in Table D.1 are also from Jensen et al. (2023) except rvol_252d, which we assign to the low-risk theme.

D.2 Portfolio Tuning

Panel A in Figure D.1 shows the optimal hyper-parameters for the RF method that predict returns. We show the results for three out of the twelve models we consider. The 1-month model is used by all methods except Portfolio-ML, while the expected return over 6 and 12 months is only used by Multiperiod-ML and Multiperiod-ML*. Panel B in Figure D.1 shows the optimal hyper-parameters for Portfolio-ML and the second layer of portfolio tuning used by Multiperiod-ML* and Static-ML*. Section 5.3 describes how we choose hyper-parameters and table 1 the range of possible hyper-parameters.

D.3 Feature Persistence and Importance across Return Horizons

Figure D.2 shows the monthly autocorrelation of all prediction features. Features grouped into themes following Jensen et al. (2023). We see that most features are highly persistent from month to month, but we also include a substantial number of fast-moving predictors. These high-frequency predictors are particularly present in the low-risk, seasonality, and short-term reversal themes. Figure D.3 shows a measure of feature importance for each of the twelve models that predict excess returns over $t+1, t+2, \ldots, t+12$. The short-term model that predicts returns one month ahead is used by all portfolio methods except Portfolio-ML, while only Multiperiod-ML uses the longer-term models. Notably, feature importance for the short-term model differs from the others by distributing importance more evenly across themes. In contrast, the longer-term models mainly use quality, value, and momentum features.

One curiosity is that short-term reversal is important for predicting returns at $t+1$, the original short-term reversal effect, but also predicts returns well at $t+12$. In unreported results, we found that the return at time t (an important characteristic within the short-term reversal theme) is negatively correlated with the predicted return at time $t+1$ but positively correlated with the predicted return at time $t+12$. As such, the importance of short-term reversal at $t+12$ likely reflects the seasonality effect documented by Heston and Sadka (2008).

[25] We add 1-year trailing volatility because of its close connection to the covariance matrix. For example, suppose volatility is unrelated to expected returns. In that case, the optimal portfolio should have a higher allocation to low volatility assets, all else equal.

Table D.1: Feature Information

	Characteristic	Theme		Characteristic	Theme		Characteristic	Theme
1	cowc_gr1a	accruals	40	ivol_capm_252d	low_risk	79	opex_at	quality
2	oaccruals_at	accruals	41	ivol_ff3_21d	low_risk	80	qmj_prof	quality
3	oaccruals_ni	accruals	42	rmax1_21d	low_risk	81	qmj_safety	quality
4	taccruals_at	accruals	43	rmax5_21d	low_risk	82	sale_bev	quality
5	taccruals_ni	accruals	44	rvol_21d	low_risk	83	corr_1260d	seasonality
6	fnl_gr1a	debt_issuance	45	turnover_126d	low_risk	84	coskew_21d	seasonality
7	ncol_gr1a	debt_issuance	46	zero_trades_126d	low_risk	85	dbnetis_at	seasonality
8	nfna_gr1a	debt_issuance	47	zero_trades_21d	low_risk	86	kz_index	seasonality
9	noa_at	debt_issuance	48	zero_trades_252d	low_risk	87	lti_gr1a	seasonality
10	aliq_at	investment	49	rvol_252d	low_risk	88	pi_nix	seasonality
11	at_gr1	investment	50	prc_highprc_252d	momentum	89	seas_11_15an	seasonality
12	be_gr1a	investment	51	ret_12_1	momentum	90	seas_11_15na	seasonality
13	capx_gr1	investment	52	ret_3_1	momentum	91	seas_2_5an	seasonality
14	coa_gr1a	investment	53	ret_6_1	momentum	92	seas_6_10an	seasonality
15	col_gr1a	investment	54	ret_9_1	momentum	93	ami_126d	size
16	emp_gr1	investment	55	seas_1_1na	momentum	94	dolvol_126d	size
17	inv_gr1	investment	56	ocf_at_chg1	profit_growth	95	market_equity	size
18	inv_gr1a	investment	57	ret_12_7	profit_growth	96	prc	size
19	lnoa_gr1a	investment	58	sale_emp_gr1	profit_growth	97	iskew_capm_21d	short-term reversal
20	mispricing_mgmt	investment	59	seas_1_1an	profit_growth	98	iskew_ff3_21d	short-term reversal
21	ncoa_gr1a	investment	60	tax_gr1a	profit_growth	99	ret_1_0	short-term reversal
22	nncoa_gr1a	investment	61	dolvol_var_126d	profitability	100	rmax5_rvol_21d	short-term reversal
23	noa_gr1a	investment	62	ebit_bev	profitability	101	rskew_21d	short-term reversal
24	ppeinv_gr1a	investment	63	ebit_sale	profitability	102	at_me	value
25	ret_60_12	investment	64	intrinsic_value	profitability	103	be_me	value
26	sale_gr1	investment	65	ni_be	profitability	104	bev_mev	value
27	seas_2_5na	investment	66	o_score	profitability	105	chcsho_12m	value
28	age	leverage	67	ocf_at	profitability	106	debt_me	value
29	aliq_mat	leverage	68	ope_be	profitability	107	div12m_me	value
30	at_be	leverage	69	ope_bel1	profitability	108	ebitda_mev	value
31	bidaskhl_21d	leverage	70	turnover_var_126d	profitability	109	eq_dur	value
32	cash_at	leverage	71	at_turnover	quality	110	eqnpo_12m	value
33	netdebt_me	leverage	72	cop_at	quality	111	fcf_me	value
34	tangibility	leverage	73	cop_atl1	quality	112	ni_me	value
35	beta_60m	low_risk	74	gp_at	quality	113	ocf_me	value
36	beta_dimson_21d	low_risk	75	gp_atl1	quality	114	sale_me	value
37	betabab_1260d	low_risk	76	mispricing_perf	quality	115	seas_6_10na	value
38	betadown_252d	low_risk	77	op_at	quality			
39	ivol_capm_21d	low_risk	78	op_atl1	quality			

Note: The table shows the security characteristics we use as features for the portfolio methods. The characteristics are from Jensen et al. (2023), and we refer to this paper for details about the construction methodology.

D.4 Portfolio Statistics over Time

Figure D.4 shows how the ex-ante volatility, leverage, and turnover evolve over time. The figures show that Portfolio-ML tends to have a lower turnover than the other methods.

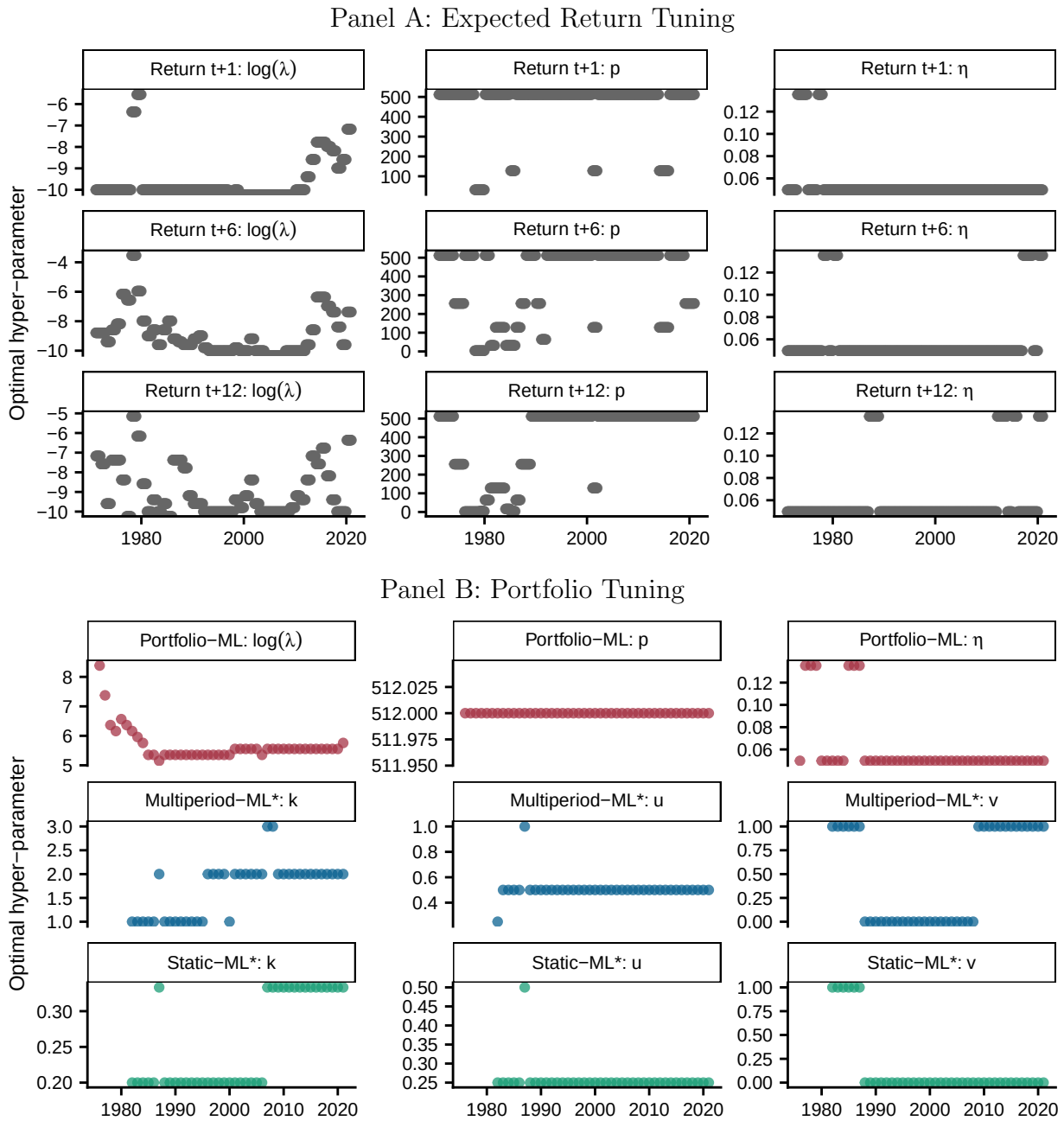

Figure D.1: Optimal Portfolio Hyper-parameters

Note: Panel A shows the optimal hyper-parameters used for predicting returns via. Ridge regression of RF transformed features (shown for three of the twelve models we consider). Panel B shows the optimal hyper-parameters for selecting portfolio weights for Portfolio-ML, Multiperiod-ML*, and Static-ML*. We show the range of possible hyper-parameters in table 1.

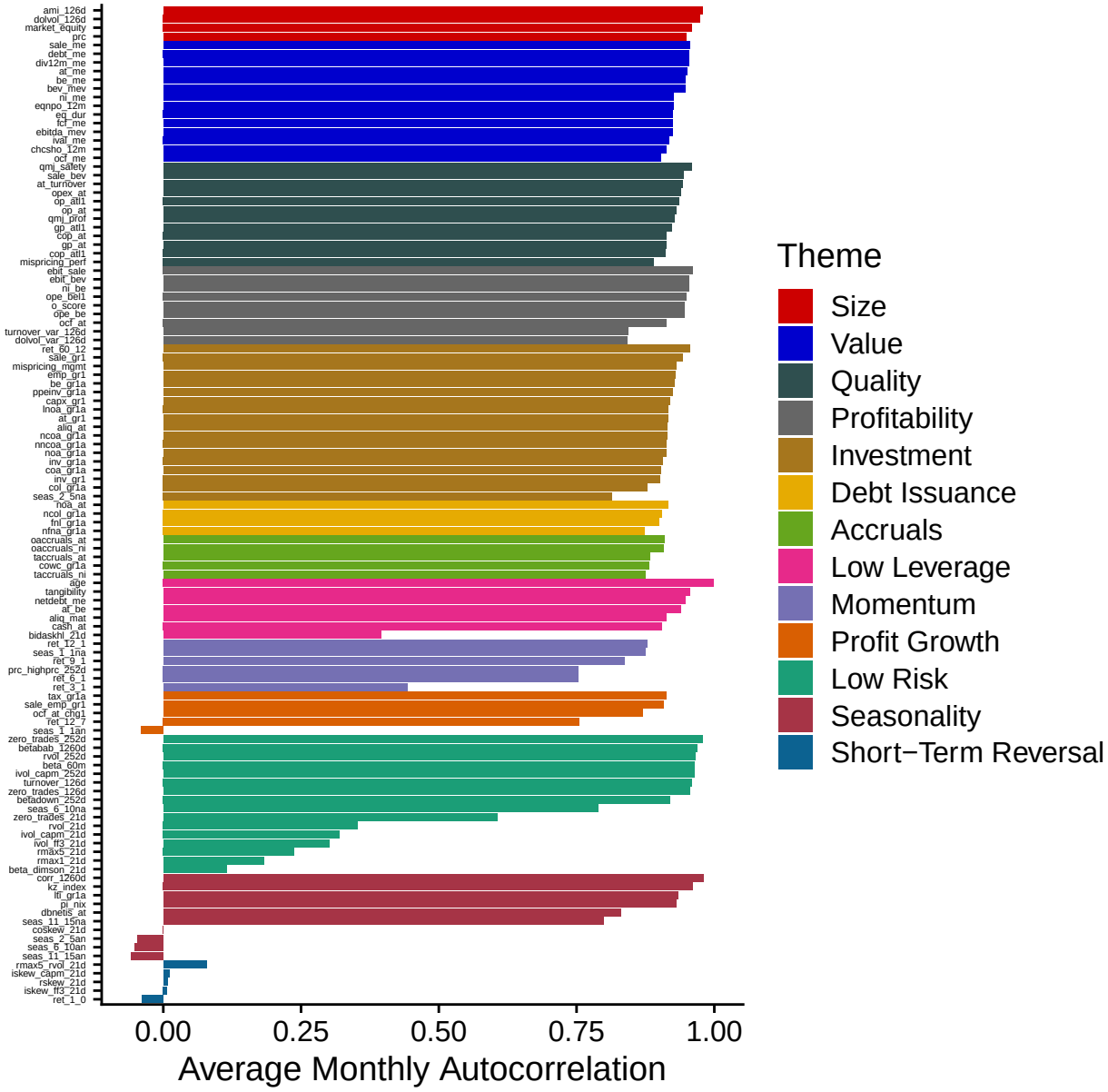

Figure D.2: Feature Autocorrelation

Note: The figure shows the monthly autocorrelation for each feature in our sample. We first compute each feature's monthly autocorrelation for all stocks with at least five years of monthly data. Next, we average the stock-level autocorrelations to arrive at the final estimate. The features are grouped by theme and sorted by average theme autocorrelation.

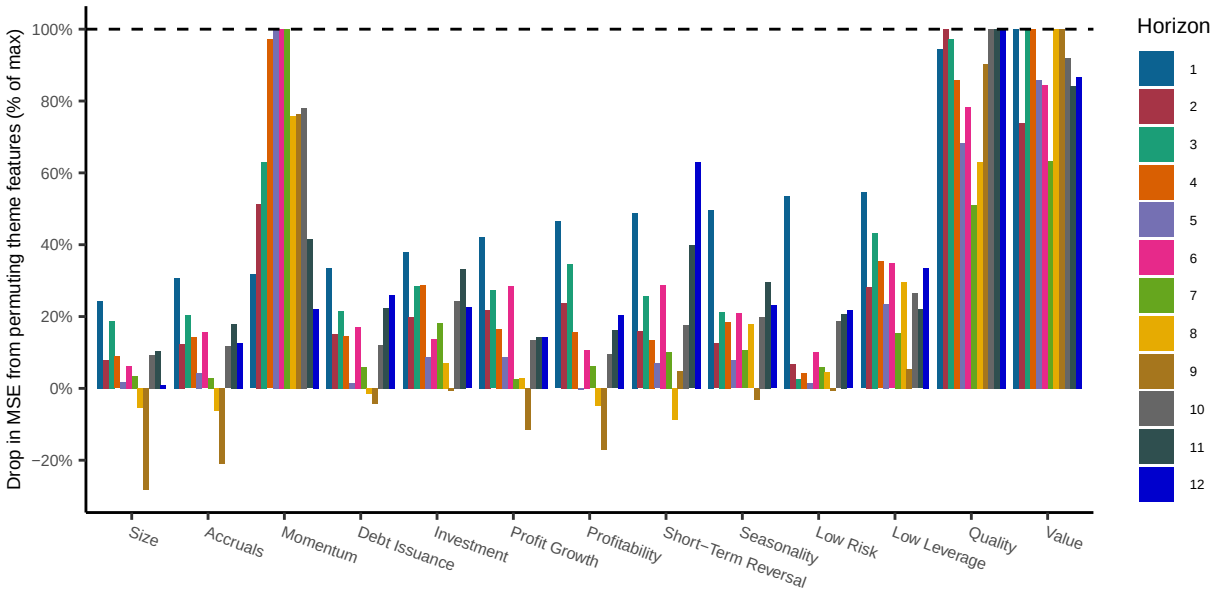

Figure D.3: Feature Importance across Return Horizons

Note: The figure shows feature importance for the twelve random feature-based models that predict returns in month $t+1, t+2, \ldots, t+12$. For each model, we randomly permute the associated features for each theme while keeping all other features at their actual value. We then implement each method based on this counterfactual data and measure feature importance as the difference in the mean-squared error relative to the implementation that uses the actual data. For comparability, we divide all values by the value for the most important feature of the specific model. Hence, feature importance is measured relative to the best feature.

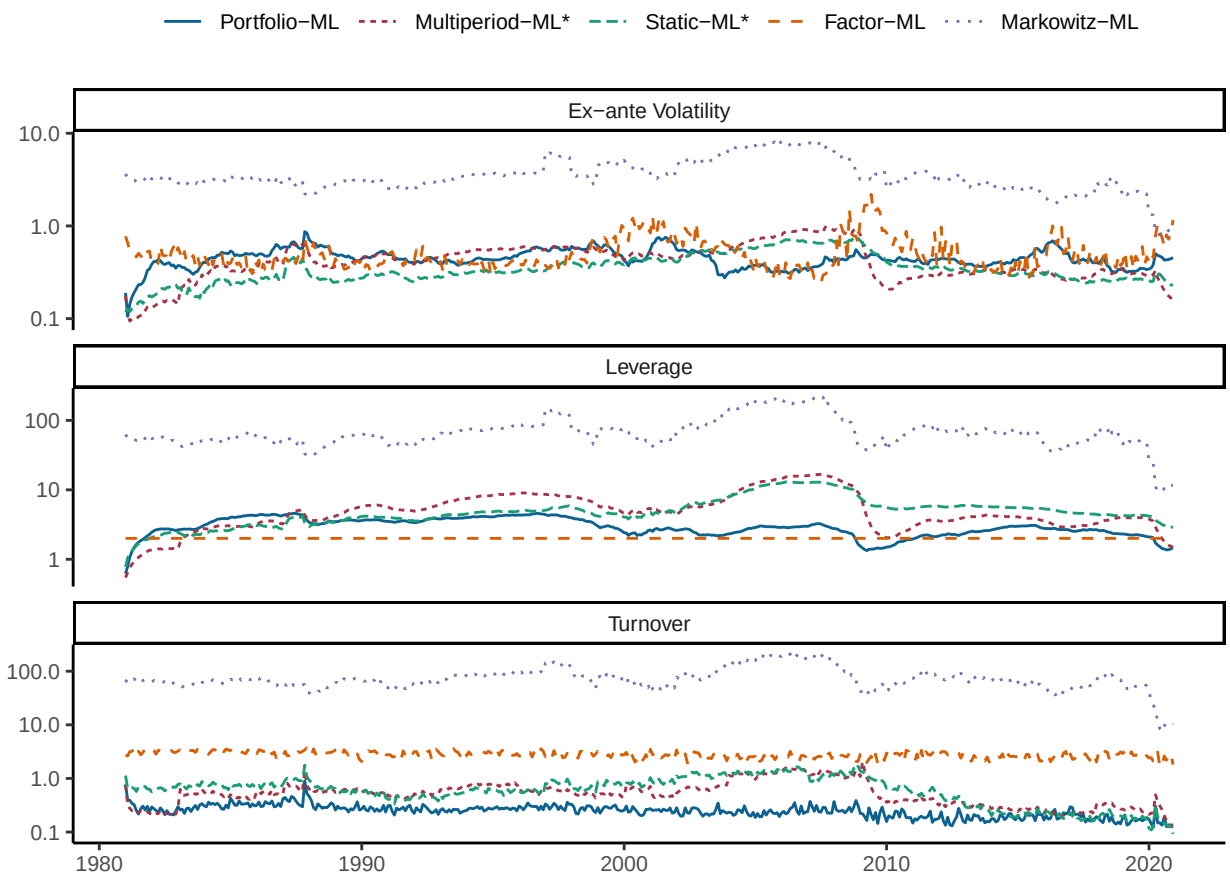

Figure D.4: Portfolio Statistics over Time

Note: The top panel shows the ex-ante volatility of each portfolio method based on a monthly updated covariance matrix. The middle panel shows portfolio leverage, which is defined as the sum of absolute portfolio weights. The lower panel shows the monthly portfolio turnover defined as $\sum_i |(\pi_t - g_t \pi_{t-1})_i|$, the sum of absolute differences between the current portfolio weight, π_t, and the grown portfolio weight from last month, $g_t \pi_{t-1}$. We use a logarithmic scale for the y-axis because of large differences across methods.

E Simulations

In this section, we simulate data to understand how the models perform under various idealized settings.

Simulation details

- *Stocks*: Balanced panel of 500 stocks observed monthly for 70 years
- *Characteristics*: For each stock, we simulate three characteristics, $x_t^i, z_{1,t}^i,$ and $z_{2,t}^i$:
  - The first, x_t^i, is drawn at the beginning of the sample from $N(1, 0.5^2)$ and remains constant throughout the sample
  - The second, $z_{1,t}^i$, is drawn at the beginning of the sample from $N(0, 0.25^2)$, and updated each period as

$$z_{1,t}^i = z_{1,t-1}^i + (1-\theta_1)(0 - z_{1,t-1}^i) + \epsilon_{1,t}^i, \qquad \epsilon_{1,t}^i \sim N(0, 0.25^2)$$

  - The third, $z_{2,t}^i$, is drawn at the beginning of the sample from $N(0, 0.25^2)$, and updated each period as

$$z_{2,t}^i = z_{2,t-1}^i + (1-\theta_2)(0 - z_{2,t-1}^i) + \epsilon_{2,t}^i, \qquad \epsilon_{2s,t}^i \sim N(0, 0.25^2)$$

- *Expected returns*: One-month ahead expected returns depends on all characteristics

$$E_t\left[r_{t+1}^i\right] = \frac{0.05}{12} x_t^i + \frac{0.025}{12}\left(z_{1,t}^i + z_{2,t}^i\right),$$

  which, along with the evolution of the characteristics, also determines expected returns over longer horizons.

- *Variance-covariance matrix*
  - Covariances are determined by x_t^i, which we assume represents loadings to a common factor (say, the market), with a variance of $\sigma_m^2 = \frac{0.2^2}{12}$.
  - Variances are also affected by idiosyncratic variance, which we assume to be constant across stocks and time at $\sigma_u^2 = \frac{0.4^2}{12}$.
  - The variance-covariance matrix is, therefore constant through time and equal to

$$\Sigma_t = \Sigma = \sigma_m^2 x_t x_t' + \text{diag}(\sigma_u^2)$$

- *Trading cost*: We assume that the daily trading volume is constant across stocks and time at \$84mil, which is the median daily dollar volume of the stocks in our investable universe by 2020.
- *Wealth*: The investor's wealth is constant over time.

Variation across simulations

We vary the investor's wealth and the persistence of the two time-varying characteristics, $z^i_{1,t}$ and $z^i_{2,t}$. Specifically, we the case where both signals are fast, $\theta_1 = \theta_2 = 0$, where one signal is fast and the other is slow, $\theta_1 = 0.95$ and $\theta_2 = 0$, and where both signals are slow $\theta_1 = \theta_2 = 0.95$. For each combination we consider five different investors with a wealth of $10^8, 10^9, \ldots, 10^{12}$. For each combination of feature persistence and investor wealth, we repeat the simulation three times.

Simulation results

Figure E.1 reports the median realized utility for each combination. Multiperiod-ML is consistently the best-performing method, followed by Static-ML for low levels of investor wealth and Portfolio-ML for high levels of investor wealth. When interpreting these results, however, it is important to note that Static-ML and Multiperiod-ML have a major advantage in that we use the true expected returns as an input. Portfolio-ML, on the other hand, has to learn the parameters from the data. Nevertheless, Portfolio-ML still performs well, especially for cases where the investor has a lot of wealth.

Since we keep the dollar trading volume fixed when we vary the investor wealth, we can also think of variation in investor wealth as variation in stock liquidity. For low investor wealth, it is as if the investor is trading very liquid stocks, and vice versa when the investor wealth is high. Therefore, the results suggest that the benefit of Multiperiod-ML and Portfolio-ML is particularly high among illiquid stocks.

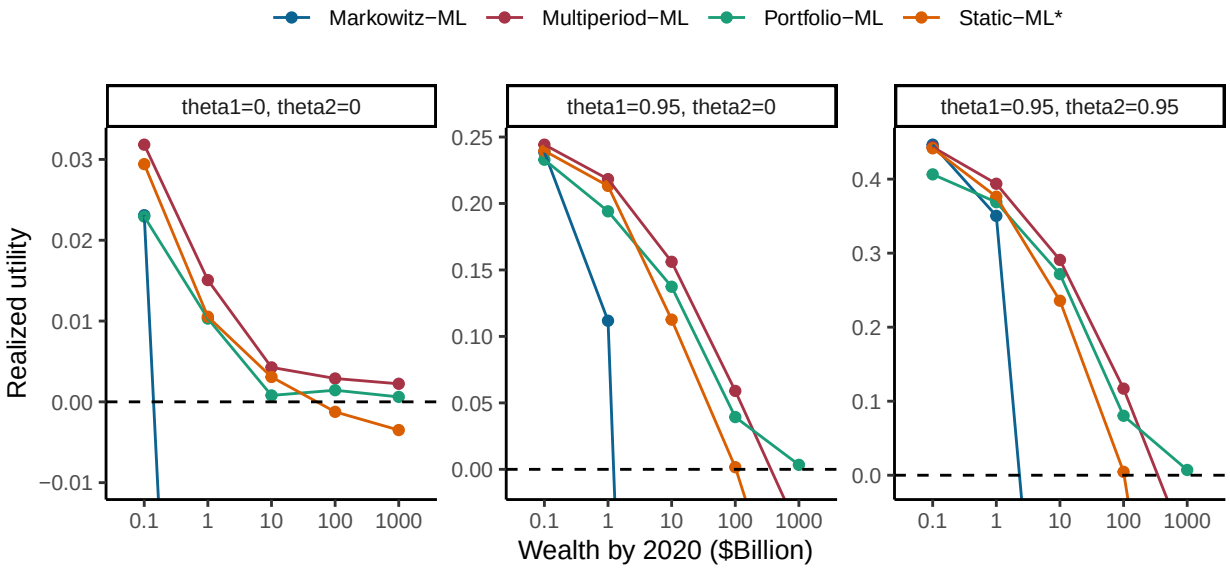

Figure E.1: Simulations

Note: The figure shows the results of implementations of various portfolio choice methods using a simulated economy. The simulated economy features 500 stocks, each with an average dollar trading volume of $84 million (the median in our data), and each stock has two time-varying characteristics, with an AR1 coefficient of theta1 and theta2 (shown in the panel title). The x-axis shows the assumed wealth of the investor. Because we assume that the dollar trading volume of stocks is fixed, we can equivalently think of variation in the investor's wealth as variation in stock liquidity (low wealth corresponds to trading very liquid stocks, and vice versa). The realized utility shown in the figure is averaged across three simulations.

Swiss Finance Institute

Swiss Finance Institute (SFI) is the national center for fundamental research, doctoral training, knowledge exchange, and continuing education in the fields of banking and finance. SFI's mission is to grow knowledge capital for the Swiss financial marketplace. Created in 2006 as a public–private partnership, SFI is a common initiative of the Swiss finance industry, leading Swiss universities, and the Swiss Confederation.

swiss:finance:institute

c/o University of Geneva, Bd. Du Pont d'Arve 42, CH-1211 Geneva 4
T +41 22 379 84 71, rps@sfi.ch, www.sfi.ch