NBER WORKING PAPER SERIES

FINANCIAL MACHINE LEARNING

Bryan T. Kelly
Dacheng Xiu

Working Paper 31502
http://www.nber.org/papers/w31502

NATIONAL BUREAU OF ECONOMIC RESEARCH
1050 Massachusetts Avenue
Cambridge, MA 02138
July 2023

AQR Capital Management is a global investment management firm, which 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. The views expressed herein are those of the authors and do not necessarily reflect the views of the National Bureau of Economic Research.

Financial Machine Learning
Bryan T. Kelly and Dacheng Xiu
NBER Working Paper No. 31502
July 2023
JEL No. C33,C4,C45,C55,C58,G1,G10,G11,G12,G17

ABSTRACT

We survey the nascent literature on machine learning in the study of financial markets. We highlight the best examples of what this line of research has to offer and recommend promising directions for future research. This survey is designed for both financial economists interested in grasping machine learning tools, as well as for statisticians and machine learners seeking interesting financial contexts where advanced methods may be deployed.

Bryan T. Kelly
Yale School of Management
165 Whitney Ave.
New Haven, CT 06511
and NBER
bryan.kelly@yale.edu

Dacheng Xiu
Booth School of Business
University of Chicago
5807 South Woodlaswn Avenue
Chicago, IL 60637
dachxiu@chicagobooth.edu

Contents

Financial Machine Learning

Bryan Kelly[1] and Dacheng Xiu[2]

[1] *Yale School of Management, AQR Capital Management, and NBER; bryan.kelly@yale.edu*
[2] *University of Chicago Booth School of Business; dacheng.xiu@chicagobooth.edu*

ABSTRACT

We survey the nascent literature on machine learning in the study of financial markets. We highlight the best examples of what this line of research has to offer and recommend promising directions for future research. This survey is designed for both financial economists interested in grasping machine learning tools, as well as for statisticians and machine learners seeking interesting financial contexts where advanced methods may be deployed.

Acknowledgements. We are grateful to Doron Avramov, Max Helman, Antti Ilmanen, Sydney Ludvigson, Yueran Ma, Semyon Malamud, Tobias Moskowitz, Lasse Pedersen, Markus Pelger, Seth Pruitt, and Guofu Zhou for helpful comments. AQR Capital Management is a global investment management firm, which 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.

1

Introduction: The Case for Financial Machine Learning

1.1 Prices are Predictions

Modern analysis of financial markets centers on the following definition of a price, derived from the generic optimality condition of an investor:

$$P_{i,t} = \mathrm{E}[M_{t+1} X_{i,t+1} | \mathcal{I}_t]. \tag{1.1}$$

In words, the prevailing price of an asset, $P_{i,t}$, reflect investors' valuation of its future payoffs, $X_{i,t+1}$. These valuations are discounted based on investors' preferences, generically summarized as future realized marginal rates of substitution, M_{t+1}. The price is then determined by investor expectations of these objects given their conditioning information $\mathcal{I}_t$. In other words, prices are predictions—they reflect investors' best guesses for the (discounted) future payoffs shed by an asset.

It is common to analyze prices in an equivalent expected return, or "discount rate," representation that normalizes (1.1) by the time t price:

$$\mathrm{E}[R_{i,t+1} | \mathcal{I}_t] = \beta_{i,t} \lambda_t, \tag{1.2}$$

where $R_{i,t+1} = X_{i,t+1}/P_{i,t} - R_{f,t}$ is the asset's excess return, $R_{f,t} = \mathrm{E}[M_{t+1} | \mathcal{I}_t]^{-1}$ is the one-period risk-free rate, $\beta_{i,t} = \frac{\mathrm{Cov}[M_{t+1}, R_{i,t+1} | \mathcal{I}_t]}{\mathrm{Var}[M_{t+1} | \mathcal{I}_t]}$ is the asset's covariance with M_{t+1}, and $\lambda_t = -\frac{\mathrm{Var}[M_{t+1}|\mathcal{I}_t]}{\mathrm{E}[M_{t+1}|\mathcal{I}_t]}$ is the price of risk. We can ask economic questions in terms of either prices or discount rates, but the literature typically opts for the discount rate representation for a few reasons. Prices are often non-stationary while discount rates are often stationary, so when the statistical properties of estimators rely on stationarity assumptions it is advantageous to work with discount rates. Also, uninteresting differences in the scale of assets' payoffs will lead to uninteresting scale differences in prices. But discount rates are typically unaffected by differences in payoff scale so the researcher need not adjust for them.

More generally, studying market phenomena in terms of returns alleviates some of the researcher's modeling burden by partially homogenizing data to have tractable dynamics and scaling properties. Besides, discount rates are also predictions, and their interpretation is especially simple and practically important. $\mathrm{E}[R_{i,t+1}|\mathcal{I}_t]$ describes investors' expectations for the appreciation in asset value over the next period. As such, the expected return is a critical input to allocation decisions. If we manage to isolate an empirical model for this expectation that closely fits the data, we have achieved a better understanding of market functionality and simultaneously derived a tool to improve resource allocations going forward. This is a fine example of duality in applied social science research: A good model both elevates scientific understanding and improves real-world decision-making.

1.2 Information Sets are Large

There are two conditions of finance research that make it fertile soil for machine learning methods: large conditioning information sets and ambiguous functional forms. Immediately evident from (1.1) is that the study of asset prices is inextricably tied to information. Guiding questions in the study of financial economics include "what information do market participants have and how do they use it?" The predictions embodied in prices are shaped by the available information that is pertinent to future asset payoffs ($X_{i,t+1}$) and investors' feelings about those payoffs (M_{t+1}). If prices behaved the same in all states of the world—e.g. if payoffs and preferences were close to i.i.d.—then information sets would drop out. But even the armchair investor dabbling in their online account or reading the latest edition of *The Wall Street Journal* quickly intuits the vast scope of conditioning information lurking behind market prices. Meanwhile, the production function of the modern asset management industry is a testament to the vast amount of information flowing into asset prices: Professional managers (in various manual and automated fashions) routinely pore over troves of news feeds, data releases, and expert predictions in order to inform their investment decisions.

The expanse of price-relevant information is compounded by the panel nature of financial markets. The price of any given asset tends to vary over time in potentially interesting ways—this corresponds to the time series dimension of the panel. Meanwhile, at a given point in time, prices differ across assets in interesting ways—the cross section dimension of the panel. Time series variation in the market environment will affect many assets in interconnected ways. For example, most asset prices behave differently in high versus low risk conditions or in different policy regimes. As macroeconomic conditions change, asset prices adjust in unison through these common effects. Additionally, there are cross-sectional behaviors that are distinct to individual assets or groups of assets. So, conditioning information is not just time series in nature, but also includes asset-level attributes. A successful model of asset behavior must simultaneously account for shared dynamic effects as well as asset-specific effects (which may themselves be static or dynamic). As highlighted by Gu *et al.* (2020b),

> *The profession has accumulated a staggering list of predictors that various researchers have argued possess forecasting power for returns. The number of stock-level predictive characteristics reported in the literature numbers in the hundreds and macroeconomic predictors of the aggregate market number in the dozens.*

Furthermore, given the tendency of financial economics research to investigate one or a few variables at a time, we have presumably left much ground uncovered. For example, only recently has the information content of news text emerged as an input to empirical models of (1.1), and there is much room for expansion on this frontier and others.

1.3 Functional Forms are Ambiguous

If asset prices are expectations of future outcomes, then the statistical tools to study prices are forecasting models. A traditional econometric approach to financial market research (e.g. Hansen and Singleton, 1982) first specifies a functional form for the return forecasting model motivated by a theoretical economic model, then estimates parameters to understand how candidate information sources associate with observed market prices within the confines of the chosen model. But which of the many economic models available in the literature should we impose?

The formulation of the first-order condition, or "Euler equation," in (1.1) is broad enough to encompass a wide variety of structural economic assumptions. This generality is warranted because there is no consensus about which specific structural formulations are viable. Early consumption-based models fail to match market price data by most measures (e.g. Mehra and Prescott, 1985). Modern structural models match price data somewhat better if the measure of success is sufficiently forgiving (e.g. Chen *et al.*, 2022a), but the scope of phenomena they describe tends to be limited to a few assets and is typically evaluated only on an in-sample basis.

Given the limited empirical success of structural models, most empirical work in the last two decades has opted away from structural assumptions to less rigid "reduced-form" or "no-arbitrage" frameworks. While empirical research of markets often steers clear of imposing detailed economic structure, it typically imposes statistical structure (for example, in the form of low-dimensional factor models or other parametric assumptions). But there are many potential choices for statistical structure in reduced-form models, and it is worth exploring the benefits of flexible models that can accommodate many different functional forms and varying degrees of nonlinearity and variable interactions.

Enter machine learning tools such as kernel methods, penalized likelihood estimators, decision trees, and neural networks. Comprised of diverse nonparametric estimators and large parametric models, machine learning methods are explicitly designed to approximate unknown data generating functions. In addition, machine learning can help integrate many data sources into a single model. In light of the discussion in Section 1.2, effective modeling of prices and expected returns requires rich conditioning information in $\mathcal{I}_t$. On this point, Cochrane (2009)[1] notes that *"We obviously don't even observe all the conditioning information used by economic agents, and we can't include even a fraction of observed conditioning information in our models."* Hansen and Richard (1987) (and more recently Martin and Nagel, 2021) highlight differences in information accessible to investors inside an economic model versus information available to an econometrician on the outside of a model looking in. Machine learning is a toolkit that can help narrow the gap between information sets of researchers and market participants by providing methods that allow the researcher to assimilate larger information sets.

The more expansive we can be in our consideration of large conditioning sets, the more realistic our models will be. This same logic applies to the question of functional form. Not only do market participants impound rich information into their forecasts, they do it in potentially complex ways that leverage the nuanced powers of human reasoning and intuition. We must recognize that investors use information in ways that we as researchers cannot know explicitly and thus cannot exhaustively (and certainly not concisely) specify in a parametric statistical model. Just as Cochrane (2009) reminds us to be circumspect in our consideration of conditioning information, we must be equally circumspect in our consideration of functional forms.

1.4 Machine Learning versus Econometrics

What is machine learning, and how is it different from traditional econometrics? Gu *et al.* (2020b) emphasize that the definition of machine learning is inchoate and the term is at times corrupted by the marketing purposes of the user. We follow Gu *et al.* (2020b) and use the term to describe (i) a diverse collection of high-dimensional models for

[1] Readers of this survey are encouraged to re-visit chapter 8 of Cochrane (2009) and recognize the many ways machine learning concepts mesh with his outline of the role of conditioning information in asset prices.

statistical prediction, combined with (ii) "regularization" methods for model selection and mitigation of overfit, and (iii) efficient algorithms for searching among a vast number of potential model specifications.

Given this definition, it should be clear that, in any of its incarnations, financial machine learning amounts to a set of procedures for estimating a statistical model and using that model to make decisions. So, at its core, machine learning *need not be differentiated* from econometrics or statistics more generally. Many of the ideas underlying machine learning have lived comfortably under the umbrella of statistics for decades (Israel *et al.*, 2020).

In order to learn through the experience of data, the machine needs a functional representation of what it is trying to learn. The researcher must make a representation choice—this is a canvas upon which the data will paint its story. Part (i) of our definition points out that machine learning brings an open-mindedness to functional representations that are highly parameterized and often nonlinear. Small models are rigid and oversimplified, but their parsimony has benefits like comparatively precise parameter estimates and ease of interpretation. Large and sophisticated models are much more flexible, but can also be more sensitive and suffer from poor out-of-sample performance when they overfit noise in the system. Researchers turn to large models when they believe the benefits from more accurately describing the complexities of real world phenomena outweigh the costs of potential overfit. At an intuitive level, machine learning is a way to pursue statistical analysis when the analyst is unsure which specific structure their statistical model should take. In this sense, much of machine learning can be viewed as nonparametric (or semi-parametric) modeling. Its modus operandi considers a variety of potential model specifications and asks the data's guidance in choosing which model is most effective for the problem at hand. One may ask: when does the analyst *ever* know what structure is appropriate for their statistical analysis? The answer of course is "never," which is why machine learning is generally valuable in financial research. As emphasized by Breiman (2001), its focus on maximizing prediction accuracy in the face of an unknown data model is the central differentiating feature of machine learning from the traditional statistical objective of estimating a known data generating model and conducting hypothesis tests.

Part (ii) of our definition highlights that machine learning chooses a preferred model (or combination of models) from a “diverse collection” of candidate models. Again, this idea has a rich history in econometrics under the heading of model selection (and, relatedly, model averaging). The difference is that machine learning puts model selection at the heart of the empirical design. The process of searching through many models to find top performers (often referred to as model “tuning”) is characteristic of all machine learning methods. Of course, selecting from multiple models mechanically leads to in-sample overfitting and can produce poor out-of-sample performance. Thus machine learning research processes are accompanied by “regularization,” which is a blanket term for constraining model size to encourage stable performance out-of-sample. As Gu *et al.* (2020b) put it, *“An optimal model is a ‘Goldilocks’ model. It is large enough that it can reliably detect potentially complex predictive relationships in the data, but not so flexible that it is dominated by overfit and suffers out-of-sample.”* Regularization methods encourage smaller models; richer models are only selected if they are likely to give a genuine boost to out-of-sample prediction accuracy.

Element (iii) in the machine learning definition is perhaps its clearest differentiator from traditional statistics, but also perhaps the least economically interesting. When data sets are large and/or models are very heavily parameterized, computation can become a bottleneck. Machine learning has developed a variety of approximate optimization routines to reduce computing loads. For example, traditional econometric estimators typically use all data points in every step of an iterative optimization routine and only cease the parameter search when the routine converges. Shortcuts such as using subsets of data and halting a search before convergence often reduce computation and do so with little loss of accuracy (see, e.g., stochastic gradient descent and early stopping which are two staples in neural network training).

1.5 Challenges of Applying Machine Learning in Finance (and the Benefits of Economic Structure)

While financial research is in many ways ideally suited to machine learning methods, some aspects of finance also present challenges for machine learning. Understanding these obstacles is important for developing realistic expectations about the benefits of financial machine learning.

First, while machine learning is often viewed as a "big data" tool, many foundational questions in finance are frustrated by the decidedly "small data" reality of economic time series. Standard data sets in macro finance, for example, are confined to a few hundred monthly observations. This kind of data scarcity is unusual in other machine learning domains where researchers often have, for all intents and purposes, unlimited data (or the ability to generate new data as needed). In time series research, new data accrues only through the passage of time.

Second, financial research often faces weak signal-to-noise ratios. Nowhere is this more evident than in return prediction, where the forces of market efficiency (profit maximization and competition) are ever striving to eliminate the predictability of price movements (Samuelson, 1965; Fama, 1970). As a result, price variation is expected to emanate predominantly from the arrival of unanticipated news (which is unforecastable noise from the perspective of the model). Markets may also exhibit inefficiencies and investor preferences may give rise to time-varying risk premia, which result in some predictability of returns. Nonetheless, we should expect return predictability to be small and fiercely competed over.

Third, investors learn and markets evolve. This creates a moving target for machine learning prediction models. Previously reliable predictive patterns may be arbitraged away. Regulatory and technological changes alter the structure of the economy. Structural instability makes finance an especially complex learning domain and compounds the challenges of small data and low signal-to-noise ratios.

These challenges present an opportunity to benefit from knowledge gained by economic theory. As noted by Israel *et al.* (2020),

> *"A basic principle of statistical analysis is that theory and model parameters are substitutes. The more structure you*

> *can impose in your model, the fewer parameters you need to estimate and the more efficiently your model can use available data points to cut through noise. That is, models are helpful because they filter out noise. But an over-simplified model can filter out some signal too, so in a data-rich and high signal-to-noise environment, you would not want to use an unnecessarily small model. One can begin to tackle small data and low signal-to-noise problems by bringing economic theory to describe some aspects of the data, complemented by machine learning tools to capture aspects of the data for which theory is silent."*

1.6 Economic Content (Two Cultures of Financial Economics)

We recall Breiman (2001)'s essay on the "two cultures" of statistics, which has an analogue in financial economics (with appropriate modifications). One is the "structural model/hypothesis test" culture, which favors imposing fully or partially specified structural assumptions and investigating economic mechanisms through hypothesis tests. The traditional program of empirical asset pricing analysis (pre-dating the emergence of reduced form factor models and machine learning prediction models) studies prices through the lens of heavily constrained prediction models. The constraints come in the form of i) specific functional forms/distributions, and ii) limited variables admitted into the conditioning information set. These models often "generalize" poorly in the sense that they have weak explanatory power for asset price behaviors outside the narrow purview of the model design or beyond the training data set. This is such an obvious statement that one rarely considers out-of-sample performance of fully specified structural asset pricing models.

The other is the "prediction model" culture, which values statistical explanatory power above all else, and is born largely from the limitations of the earlier established structural culture. The prediction model culture willingly espouses model specifications that might lack an explicit association with economic theory, so long as they produce meaningful, robust improvements in data fit versus the status quo. In addition to reduced-form modeling that has mostly dominated empirical finance since the 1990's, financial machine learning research to date falls squarely in this second culture.

"There is no economics" is a charge sometimes lobbed at the statistical prediction research by economic seminar audiences, discussants, and referees. This criticism is often misguided and we should guard against it unduly devaluing advancements in financial machine learning. Let us not miss the important economic role of even the purest statistical modeling applications in finance. Relatively unstructured prediction models makes them no less economically important than the traditional econometrics of structural hypothesis testing, they just play a different scientific role. Hypothesis testing learns economics by probing specific economic mechanisms. But economics is not just about testing theoretical mechanisms. Atheoretical (for lack of a better term) prediction models survey the empirical landscape in broader terms, charting out new empirical facts upon which theories can be developed, and for which future hypothesis tests can investigate mechanisms. These two forms of empirical investigation—precision testing and general cartography—play complementary roles in the Kuhnian process of scientific advancement.

Consider the fundamental question of asset pricing research: What determines asset risk premia? Even if we could observe expected returns perfectly, we would still need theories to explain their behavior and empirical analysis to test those theories. But we can't observe risk premia, and they are stubbornly hard to estimate. Machine learning makes progress on measuring risk premia, which facilitates development of better theories of economic mechanisms that determine their behavior.

A critical benefit of expanding the set of known contours in the empirical landscape is that, even if details of the economic mechanisms remain shrouded, economic actors—financial market participants in particular—can always benefit from improved empirical maps. The prediction model culture has a long tradition of producing research to help investors, consumers, and policymakers make better decisions. Improved predictions provide more accurate descriptions of the state-dependent distributions faced by these economic actors.

Economics is by and large an applied field. The economics of the prediction model culture *lies precisely in* its ability to improve predictions.

Armed with better predictions—i.e., more accurate assessments of the economic opportunity set—agents can better trade off costs and benefits when allocating scarce resources. This enhances welfare. Nowhere is this more immediately clear than in the portfolio choice problem. We may not always understand the economic mechanisms by which a model delivers better return or risk forecasts; but if it does, it boosts the utility of investors and is thus economically important.

Breiman's central criticism of the structural hypothesis test culture is that

> *"when a model is fit to data to draw quantitative conclusions: the conclusions are about the model's mechanism, and not about nature's mechanism. If the model is a poor emulation of nature, the conclusions may be wrong."*

We view this less as a criticism of structural modeling, which must remain a foundation of empirical finance, but rather as a motivation and defense of prediction models. The two-culture dichotomy is, of course, a caricature. Research spans a spectrum and draws on multiple tools, and researchers do not separate into homogenous ideological camps. Both cultures are economically important. Breiman encourages us to consider flexible, even nonparametric, models to learn about economic mechanisms:

> *"The point of a model is to get useful information about the relation between the response and predictor variables. Interpretability is a way of getting information. But a model does not have to be simple to provide reliable information about the relation between predictor and response variables; neither does it have to be a [structural] data model."*

Prediction models are a first step to understanding mechanisms. Moreover, structural modeling can benefit directly from machine learning without sacrificing pointed hypothesis tests or its specificity of economic mechanisms.[2] Thus far machine learning has predominantly served the prediction model culture of financial economics. It is important to recognize it as a similarly potent tool for the structural hypothesis testing

[2] See, for example, our discussion of Chen and Ludvigson (2009), in Section 5.5.

culture (this is a critical direction for future machine learning research in finance). Surely, a research program founded solely on “measurement without theory” (Koopmans, 1947) is better served by also considering data through the lens of economic theory and with a deep understanding of the Lucas Jr, 1976 critique. Likewise, a program that only interprets data through extant economic models can overlook unexpected yet economically important statistical patterns.

Hayek (1945) confronts the economic implications of dispersed information for resource allocation. Regarding his central question of how to achieve an effective economic order, he notes

> *If we possess all the relevant information, if we can start out from a given system of preferences, and if we command complete knowledge of available means, the problem which remains is purely one of logic... This, however, is emphatically* not *the economic problem which society faces. And the economic calculus which we have developed to solve this logical problem, though an important step toward the solution of the economic problem of society, does not yet provide an answer to it. The reason for this is that the ‘data’ from which the economic calculus starts are never for the whole society ‘given’ to a single mind which could work out the implications and can never be so given.*

While Hayek’s main interest is in the merits of decentralized planning, his statements also have implications for information technologies in general, and prediction technologies in particular. Let us be so presumptuous as to reinterpret Hayek’s statement as a statistical problem: There is a wedge between the efficiency of allocations achievable by economic agents when the data generating process (DGP) is known, versus when it must be estimated. First, there is the problem of model specification—economic agents simply cannot be expected to correctly specify their statistical models. They must use some form of mis-specified parametric model or a nonparametric approximating model. In either case, mis-specification introduces a wedge between the optimal allocations achievable when the DGP is known (call this “first-best”) and the allocations derived from their mis-specified models (call this “second-best”).

But even second best is implausible, because we must estimate these models with finite data. This gives rise to yet another wedge, that due to sampling variation. Even if we knew the functional form of the DGP, we still must estimate it and noise in our estimates produces deviations from first-best. Compound that with the realism of mis-specification, and we recognize that in reality we must always live with "third-best" allocations; i.e., mis-specified models that are noisily estimated.

Improved predictions derived from methods that can digest vast information and data sets provide an opportunity to mitigate the wedges between the pure "logic" problem of first-best resource allocation noted by Hayek, and third-best realistic allocations achievable by economic agents. The wedges never shrink to zero due to statistical limits to learnability (Da *et al.*, 2022; Didisheim *et al.*, 2023). But powerful approximating models and clever regularization devices mean that machine learning is economically important exactly because they can lead to better decisions. The problem of portfolio choice is an illustrative example. A mean-variance investor who knows the true expected return and covariance matrix of assets simply executes the "logic" of a Markowitz portfolio and achieves a first-best allocation. But, in analogy to Hayek, this is emphatically *not* the problem that real world investors grapple with. Instead, their problem is primarily one of measurement—one of prediction. The investor seeks a sensible expected return and covariance estimate that, when combined with the Markowitz objective, performs reasonably well out-of-sample. Lacking high-quality measurements, the Markowitz solution can behave disastrously, as much research has demonstrated.

2

The Virtues of Complex Models

Many of us acquired our econometrics training in a tradition anchored to the "principle of parsimony." This is exemplified by the Box and Jenkins (1970) model building methodology, whose influence on financial econometrics cannot be overstated. In the introduction to the most recent edition of the Box and Jenkins forecasting textbook,[1] "parsimony" is presented as the first of the "Basic Ideas in Model Building": *"It is important, in practice, that we employ the* smallest possible *number of parameters for adequate representations"* [their emphasis].

The principle of parsimony appears to clash with the massive parameterizations adopted by modern machine learning algorithms. The leading edge GPT-3 language model (Brown *et al.*, 2020) uses 175 billion parameters. Even the comparatively skimpy return prediction neural networks in Gu *et al.* (2020b) have roughly 30,000 parameters. For an econometrician of the Box-Jenkins lineage, such rich parameterizations seem profligate, prone to overfit, and likely disastrous for out-of-sample performance.

Recent results in various non-financial domains contradict this view. In applications such as computer vision and natural language processing,

[1] Box *et al.* (2015), fifth edition of the original Box and Jenkins (1970).

models that carry astronomical parameterizations—and that *exactly fit* training data—are often the best performing models out-of-sample. In sizing up the state of the neural network literature, Belkin (2021) concludes *"it appears that the largest technologically feasible networks are consistently preferable for best performance."* Evidently, modern machine learning has turned the principle of parsimony on its head.

The search is underway for a theoretical rationale to explain the success of behemoth parameterizations and answer the question succinctly posed by Breiman (1995): *"Why don't heavily parameterized neural networks overfit the data?"* In this section we offer a glimpse into the answer. We draw on recent advances in the statistics literature which characterize the behavior of "overparameterized" models (those whose parameters vastly outnumber the available training observations).[2]

Recent literature has taken the first step of understanding the *statistical* theoretical implications of machine learning models and focus on out-of-sample prediction accuracy of overparameterized models. In this section, we are interested in the *economic* implications of overparameterization and overfit in financial machine learning. A number of finance papers have documented significant empirical gains in machine learning return prediction. The primary economic use case of these predictions is constructing utility-optimizing portfolios. We orient our development toward understanding the out-of-sample risk-return tradeoff of "machine learning portfolios" derived from highly parameterized return prediction models. Our development closely follows Kelly *et al.* (2022a) and Didisheim *et al.* (2023).

2.1 Tools For Analyzing Machine Learning Models

Kelly *et al.* (2022a) propose a thought experiment. Imagine an analyst seeking a successful return prediction model. The asset return R is generated by a true model of the form

$$R_{t+1} = f(X_t) + \epsilon_{t+1} \tag{2.1}$$

[2] This line of work falls under headings like "overparameterization," "benign overfit," and "double descent," and includes (among others) Spigler *et al.* (2019), Belkin *et al.* (2018), Belkin *et al.* (2019), Belkin *et al.* (2020), Bartlett *et al.* (2020), Jacot *et al.* (2018), Hastie *et al.* (2019), and Allen-Zhu *et al.* (2019).

where the set of predictor variables X may be known to the analyst, but the true prediction function f is unknown. Absent knowledge of f and motivated by universal approximation theory (e.g., Hornik *et al.*, 1990), the analyst decides to approximate f with a basic neural network:

$$f(X_t) \approx \sum_{i=1}^{P} S_{i,t}\beta_i.$$

Each feature in this regression is a pre-determined nonlinear transformation of raw predictors,[3]

$$S_{i,t} = \tilde{f}(w_i' X_t). \tag{2.2}$$

Ultimately, the analyst estimates the approximating regression

$$R_{t+1} = \sum_{i=1}^{P} S_{i,t}\beta_i + \tilde{\epsilon}_{t+1}. \tag{2.3}$$

The analyst has T training observations to learn from and must choose how rich of an approximating model to use—she must choose P. Simple models with small P enjoy low variance, but complex models with large P (perhaps even $P > T$) can better approximate the truth. What level of model complexity (which P) should the analyst opt for?

Perhaps surprisingly, Kelly *et al.* (2022a) show that the analyst should use the largest approximating model that she can compute! Expected out-of-sample forecasting and portfolio performance are *increasing* in model complexity.[4] To arrive at this answer, Kelly *et al.* (2022a) work with two key mathematical tools for analyzing complex nonlinear (i.e., machine learning) models. These are ridge regression with generated nonlinear features (just like $S_{i,t}$ above) and random matrix theory for dealing with estimator behavior when P is large relative to the number of training data observations.

[3] Our assumption that weights w_i and nonlinear function $\tilde{f}$ are known follows Rahimi and Recht (2007), who prove that universal approximation results apply even when the weights are randomly generated.

[4] This is true without qualification in the high complexity regime ($P > T$) and is true in the low complexity regime as long as appropriate shrinkage is employed. Kelly *et al.* (2022a) derive the choice of shrinkage that maximizes expected out-of-sample model performance.

2.1.1 Ridge Regression with Generated Features

Their first modeling assumption[5] focuses on high-dimensional linear prediction models according to (2.3), which we refer to as the "empirical model." The interpretation of (2.3) is *not* that asset returns are subject to a large number of linear fundamental driving forces. It is instead that the DGP is unknown, but may be approximated by a *nonlinear* expansion S of some (potentially small) underlying driving variables, X.[6] In the language of machine learning, S are "generated features" derived from the raw features X, for example via nonlinear neural network propagation.

A definitive aspect of this problem is that the empirical model is mis-specified. Correct specification of (2.3) requires an infinite series expansion, but in reality we are stuck with a finite number of terms, P. Small P models are stable because there are few parameters to estimate (low variance), but provide a poor approximation to the truth (large bias). A foundational premise of machine learning is that flexible (large P) model specifications can be leveraged to improve prediction. Their estimates may be noisy (high variance) but provide a more accurate approximation (small bias). It is not obvious, ex ante, which choices of P are best in terms of the bias-variance tradeoff. As economists, we ultimately seek a bias-variance tradeoff that translates into optimal economic outcomes—like higher utility for investors. The search for models that achieve economic optimality guides Kelly *et al.* (2022a)'s theoretical pursuit of high complexity models.

The second modeling assumption chooses the estimator of (2.3) to be ridge-regularized least squares:

$$\widehat{\beta}(z) = \left(zI + T^{-1} \sum_t S_t S_t' \right)^{-1} \frac{1}{T} \sum_t S_t R_{t+1}, \tag{2.4}$$

[5] We introduce assumptions here at a high level. We refer interested readers to Kelly *et al.* (2022a) for detailed technical assumptions.

[6] As suggested by the derivation of (2.3) from equation (2.1), this sacrifices little generality as a number of recent papers have established an equivalence between high-dimensional linear models and more sophisticated models such as deep neural networks (Jacot *et al.*, 2018; Hastie *et al.*, 2019; Allen-Zhu *et al.*, 2019). For simplicity, Kelly *et al.* (2022a) focus on time-series forecasts for a single risky asset (extended to multi-asset panels by Didisheim *et al.* (2023)).

where z is the ridge shrinkage parameter. Not all details of this estimator are central to our arguments—but regularization is critical. Without regularization the denominator of (2.4) is singular in the high complexity regime ($P > T$), though we will see it also has important implications for the behavior of $\widehat{\beta}(z)$ in low complexity models ($P < T$).

Finally, to characterize the economic consequences of high complexity models for investors, Kelly *et al.* (2022a) assume that investors use predictions to construct a trading strategy defined as

$$R^{\pi}_{t+1} = \pi_t R_{t+1},$$

where π_t is a timing weight that scales the risky asset position up and down in proportion to the model's return forecast. For their analysis, π_t is set equal to the expected out-of-sample return from their complex prediction model. And they assume investor well-being is measured in terms of the unconditional Sharpe ratio,

$$SR = \frac{\mathrm{E}[R^{\pi}_{t+1}]}{\sqrt{\mathrm{E}[(R^{\pi}_{t+1})^2]}}. \tag{2.5}$$

While there are other reasonable trading strategies and performance evaluation criteria, these are common choices among academics and practitioners and have the benefit of transparency and tractability.

2.1.2 Random Matrix Theory

The ridge regression formulation above couches machine learning models like neural networks in terms of linear regression. The hope is that, with this representation, it may be possible to say something concrete about expected out-of-sample behaviors of complicated models in the $P \to \infty$ limit with $P/T \to c > 0$. The necessary asymptotics for machine learning are different than those for standard econometric characterizations (which use asymptotic approximations for $T \to \infty$ with P fixed). Random matrix theory is ideally suited for describing the behavior of ridge regression in the large P setting. To simplify notation, we eliminate T from the discussion by always referring to the degree of model parameterization relative to the amount of training data. That is, we will simply track the ratio $c = P/T$, which we refer to as "model complexity."

The crux of characterizing $\widehat{\beta}(z)$ behavior when $P \to \infty$ is the $P \times P$ sample covariance matrix of signals, $\widehat{\Psi} := T^{-1} \sum_t S_t S_t'$. Random matrix theory describes the limiting distribution of $\widehat{\Psi}$'s eigenvalues. Knowledge of this distribution is sufficient to pin down the expected out-of-sample prediction performance (R^2) of ridge regression, as well as the expected out-of-sample Sharpe ratio of the associated timing strategy. More specifically, these quantities are determined by

$$m(z; c) := \lim_{P \to \infty} \frac{1}{P} \operatorname{tr}\left((\widehat{\Psi} - zI)^{-1}\right) \tag{2.6}$$

which is the limiting Stieltjes transform of the eigenvalue distribution of $\widehat{\Psi}$. From (2.6), we recognize a close link to ridge regression because the Stieltjes transform involves the ridge matrix $(\widehat{\Psi} - zI)^{-1}$. The functional form of $m(z; c)$ is known from a generalized version of the Marcenko-Pastur law. From $m(z; c)$ we can explicitly calculate the expected out-of-sample R^2 and Sharpe ratio and its sensitivity to the prediction model's complexity (see Sections 3 and 4 of Kelly *et al.* (2022a) for a detailed elaboration of this point).

In other words, model complexity plays a critical role in understanding model behavior. If T grows at a faster rate than the number of predictors (i.e., $c \to 0$), traditional large T and fixed P asymptotics kick in. In this case the expected out-of-sample behavior of a model coincides with the behavior estimated in-sample. Naturally, this is an unlikely and fairly uninteresting scenario. The interesting case corresponds to highly parameterized machine learning models with scarce data, $P/T \to c > 0$, and it is here that surprising out-of-sample model behaviors emerge.

2.2 Bigger Is Often Better

Kelly *et al.* (2022a) provide rigorous theoretical statements about the properties of high complexity machine learning models and associated trading strategies. Our current exposition focuses on the main qualitative aspects of those results based on their calibration to the market return prediction problem. In particular, they assume total return volatility equal to 20% per year and an infeasible "true" predictive R^2 of 20% per month (if the true functional form and all signals were fully available to the forecaster). Complexity hinders the model's ability to hone in

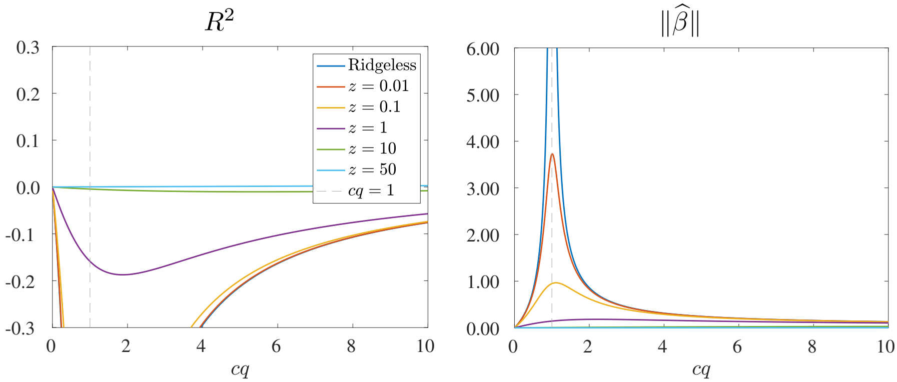


Figure 2.1: Expected Out-of-sample Prediction Accuracy From Mis-specified Models

Note: Limiting out-of-sample R^2 and $\widehat{\beta}$ norm of ridge regression as a function of c and z from Proposition 5 of Kelly *et al.* (2022a). Calibration assumes Ψ is the identity matrix, $b_* = 0.2$, and the complexity of the true model is 10.

on the true DGP because there is not enough data to support the model's heavy parameterization, thus the best *feasible* R^2 implied by this calibration is closer to 1% per month. We focus on the mis-specified setting by considering empirical models that use various subsets of the true predictors.[7]

In this calibration, the true unknown DGP is assumed to have a complexity of $c = 10$. The parameter $q \in [0, 1]$ governs the complexity of the empirical model relative to the truth. We analyze the behavior of approximating empirical models that range in complexity from very simple ($q \approx 0,\ cq \approx 0$ and thus severely mis-specified) to highly complex ($q = 1,\ cq = 10$ corresponds to the richest approximating model and in fact recovers the correct specification). Models with very low complexity are poor approximating models but their parameters can be estimated with precision. As cq rises, the empirical model better approximates the truth, but forecast variance rises (if not regularized). The calibration also considers a range of ridge penalties, z.

First consider the ordinary least squares (OLS) estimator, $\widehat{\beta}(0)$, which is a special case of (2.4) with $z = 0$. When $c \approx 0$ the model is

[7] For simplicity the predictors are assumed to be exchangeable, so only the size of a subset matters and not the identity of the specific predictors within the subset.

very simple, so it has no power to approximate the truth and delivers an R^2 of essentially zero. As P rises and approaches T from below, the model's approximation improves, but the denominator of the least squares estimator blows up, creating explosive forecast error variance. This phenomenon is illustrated in Figure 2.1. When $P = T$, the model exactly fits, or "interpolates," the training data (for this reason, $c = 1$ is called the "interpolation boundary"), so a common interpretation of the explosive behavior of $\widehat{\beta}(0)$ is severe overfit that does not generalize out-of-sample.

As P moves beyond T we enter the overparameterized, or high complexity, regime. In this regime, there are more parameters than observations, the least squares problem has multiple solutions, and the inverse covariance matrix of regressors is not defined. However, its pseudo-inverse is defined, and this corresponds to a particular unique solution to the least squares problem: $\left(T^{-1}\sum_t S_t S_t'\right)^+ \frac{1}{T}\sum_t S_t R_{t+1}$.[8] Among the many solutions that exactly fit the training data, this one has the smallest ℓ_2 norm. In fact, it is equivalent to the ridge estimator as the shrinkage parameter approaches zero:

$$\widehat{\beta}(0^+) = \lim_{z\to 0+}\left(zI + T^{-1}\sum_t S_t S_t'\right)^{-1}\frac{1}{T}\sum_t S_t R_{t+1}.$$

The solution $\widehat{\beta}(0^+)$ is called the "ridgeless" regression estimator (the blue line in the Figure 2.1). When $c \leq 1$, OLS is the ridgeless estimator, while for $c > 1$ the ridgeless case is defined by the limit $z \to 0$.

Surprisingly, the ridgeless R^2 *rises* as model complexity increases above 1. The reason is that, as c becomes larger, there is a larger space of solutions for ridgeless regression to search over and thus it can find betas with smaller ℓ_2 norm that still interpolate the training data. This acts as a form of shrinkage, biasing the beta estimate toward zero. Due to this bias, the forecast variance drops and the R^2 improves. In other words, despite $z \to 0$, the ridgeless solution still regularizes the least squares estimator, and more so the larger is c. By the time c is very large, the expected out-of-sample R^2 moves into positive territory. This property

[8] Recall that the Moore-Penrose pseudo-inverse A^+ of a matrix A is defined via $A^+ = (A'A)^{-1}A'$ if $A'A$ is invertible, and $A^+ = A'(AA')^{-1}$ if AA' is invertible.

of ridgeless least squares is a newly documented phenomenon in the statistics literature and is still an emerging topic of research.[9] The result challenges the standard financial economics doctrine that places heavy emphasis on model parsimony, demonstrating that one can improve the accuracy of return forecasts by pushing model dimensionality well beyond sample size.[10]

Figure 2.1 describes the *statistical* behavior of high complexity models. Figure 2.2 turns attention to their *economic* consequences. The right panel shows volatility of the machine learning trading strategy as a function of model complexity. The strategy's volatility moves one for one with the norm of $\widehat{\beta}$ and with the R^2 (all three quantities are different representations of forecast error variance). The important point is that, as model complexity increases past $c = 1$, trading strategy volatility continually decreases. Complexity raises the implicit shrinkage of the ridgeless estimator, which reduces return volatility (and $z > 0$ reduces volatility even further).

The left panel of Figure 2.2 shows the key economic behavior of high complexity models—the out-of-sample expected return of the timing strategy. Expected returns are low for simple strategies. Again, this is because simple models give a poor approximation of the DGP. Increasing model complexity gets you closer to the truth and monotonically increases trading strategy expected returns.[11]

What does this mean for investor well-being? The bottom panel in Figure 2.2 shows utility in terms of expected out-of-sample Sharpe ratio.[12] Out-of-sample Sharpe ratios boil down to a classic bias-variance tradeoff. Expected returns purely capture bias effects. For low complexity

[9] See Spigler *et al.* (2019), Belkin *et al.* (2018), Belkin *et al.* (2019), Belkin *et al.* (2020), and Hastie *et al.* (2019).

[10] The remaining curves in Figure 2.1 show how the out-of-sample R^2 is affected by non-trivial ridge shrinkage. The basic R^2 patterns are the same as the ridgeless case, but allowing $z > 0$ can improve R^2 further.

[11] In the ridgeless case, the benefit of added complexity reaches its maximum when $c = 1$. The ridgeless expected return is flat for $c \geq 1$ because the incremental improvements in DGP approximation are exactly offset by the gradually rising bias of ridgeless shrinkage.

[12] In the calibration, the buy-and-hold trading strategy is normalized to have a Sharpe ratio of zero. Thus, the Sharpe ratio in Figure 2.2 is in fact the Sharpe ratio *gain* from timing based on model forecasts, relative to a buy-and hold investor.

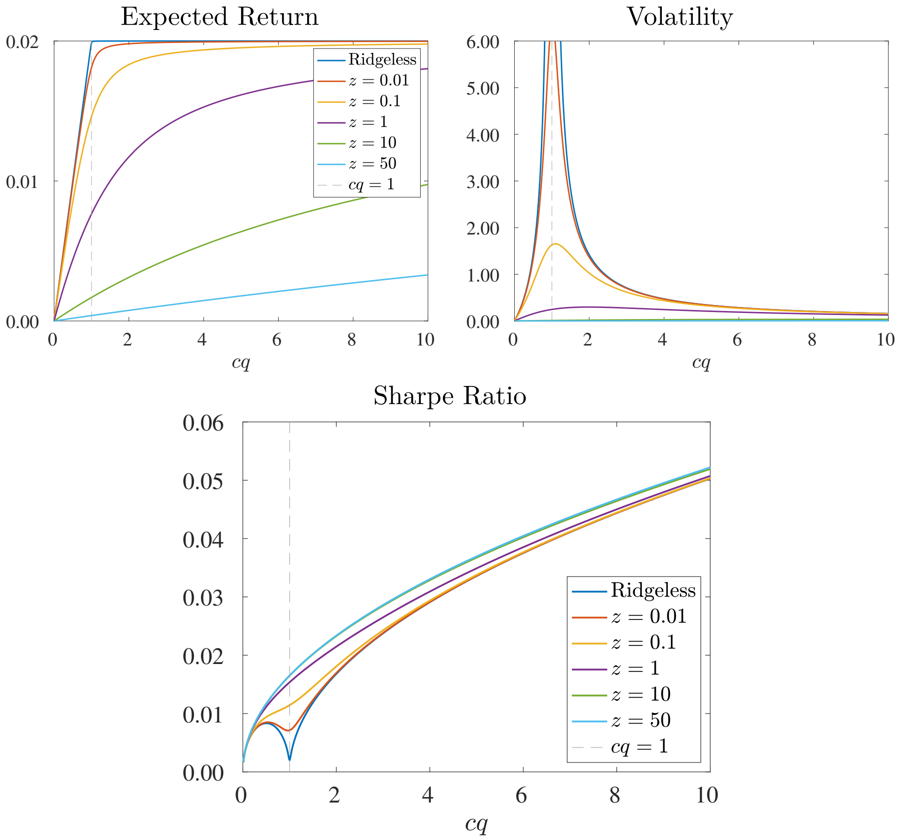


Figure 2.2: Expected Out-of-sample Risk-Return Tradeoff of Timing Strategy

Note: Limiting out-of-sample expected return, volatility, and Sharpe ratio of the timing strategy as a function of cq and z from Proposition 5 of Kelly *et al.* (2022a). Calibration assumes Ψ is the identity matrix, $b_* = 0.2$, and the complexity of the true model is $c = 10$.

models, bias comes through model misspecification but *not* through shrinkage. For high complexity models, the mis-specification bias is small, but the shrinkage bias becomes large. The theory, which shows that expected returns rise with complexity, reveals that misspecification bias is more costly than shrinkage bias when it comes to expected returns. Meanwhile, the strategy's volatility is purely about forecast variance effects. Both simple models ($c \approx 0$) or highly complex models

($c \gg 1$) produce low variance. Given these patterns in the bias-variance tradeoff, it follows that the out-of-sample Sharpe ratio is also increasing in complexity, as shown in Figure 2.2.

It is interesting to compare these findings with the phenomenon of "double descent," or the fact that out-of-sample MSE has a non-monotonic pattern in model complexity when z is close to zero (Belkin *et al.*, 2018; Hastie *et al.*, 2019). The mirror image of double descent in MSE is "double ascent" of the ridgeless Sharpe ratio. Kelly *et al.* (2022a) prove that the ridgeless Sharpe ratio dip at $c = 1$ is an artifact of insufficient shrinkage. With the right amount of shrinkage (explicitly characterized by Kelly *et al.* (2022a)), complexity becomes a virtue even in the low complexity regime: the hump disappears, and "double ascent" turns into "permanent ascent."

In summary, these results challenge the dogma of parsimony discussed at the start of this section. They demonstrate that, in the realistic case of mis-specified empirical models, complexity is a virtue. This is true not just in terms of out-of-sample statistical performance (as shown by Belkin *et al.*, 2019; Hastie *et al.*, 2019, and others) but also in the economic terms of out-of-sample investor utility. Contrary to conventional wisdom, the performance of machine learning portfolios can be theoretically improved by pushing model parameterization far beyond the number of training observations.

Kelly *et al.* (2022a) conclude with a recommendation for best practices in the use of complex models:

> *"Our results are* not *a license to add arbitrary predictors to a model. Instead, we encourage i) including all plausibly relevant predictors and ii) using rich nonlinear models rather than simple linear specifications. Doing so confers prediction and portfolio benefits, even when training data is scarce, and particularly when accompanied by prudent shrinkage."*

To derive the results above, Kelly *et al.* (2022a) impose an assumption that predictability is uniformly distributed across the signals. At first glance, this may seem overly restrictive, as many standard predictors would not satisfy this assumption. However, the assumption is consistent with (and is indeed motivated by) standard neural network models in which raw features are mixed and nonlinearly propagated into final generated features, as in equation (2.2). The ordering of the generated features S is essentially randomized by the initialization step of network training. Furthermore, in their empirical work, Kelly *et al.* (2022a), Kelly *et al.*, 2022b, and Didisheim *et al.* (2023) use a neural network formulation known as random feature regression that ensures this assumption is satisfied.

2.3 The Complexity Wedge

Didisheim *et al.* (2023) extend the analysis of Kelly *et al.* (2022a) in a number of ways and introduce the "complexity wedge," defined as the expected difference between in-sample and out-of-sample performance. For simplicity, consider a correctly specified empirical model. In a low complexity environment with $c \approx 0$, the law of large numbers applies, and as a result in-sample estimates converge to the true model. Because of this convergence, the model's in-sample performance is exactly indicative of its expected out-of-sample performance. That is, with no complexity, there is no wedge between in-sample and out-of-sample behavior.

But when $c > 0$, a complexity wedge emerges, consisting of two components. Complexity inflates the trained model's in-sample predictability relative to the true model's predictability—this is the traditional definition of overfit and it is the first wedge component. But high complexity also means that the empirical model does not have enough data (relative to its parameterization) to recover the true model—this is a failure of the law of large numbers due to complexity. This gives rise to the second wedge component, which is a shortfall in out-of-sample performance relative to the true model. This shortfall can be thought of as the "limits to learning" due to model complexity. The complexity wedge—the expected difference between in-sample and out-of-sample performance—is the sum of overfit and limits to learning.

The complexity wedge has a number of intriguing implications for asset pricing. Given a realized (feasible) prediction R^2, one can use random matrix theory to back out the extent of predictability in the "true" (but infeasible) model. A number of studies have documented significantly positive out-of-sample return prediction using machine learning models, on the order of roughly 1% per month for individual stocks. This fact, combined with a theoretical derivation for the limits to learning, suggest that the true infeasible predictive R^2 must be much higher. Relatedly, even if there are arbitrage (or simply very high Sharpe ratio) opportunities implied by the true model, limits to learning make these inaccessible to real-world investors. In a realistic empirical setting, Didisheim *et al.* (2023) suggest that attainable Sharpe ratios are attenuated by roughly an order of magnitude relative to the true data-generating process due to the infeasibility of accurately estimating complex statistical relationships.

Da *et al.* (2022) consider a distinct economic environment in which agents, namely arbitrageurs, adopt statistical arbitrage strategies in efforts to maximize their out-of-sample Sharpe ratio. These arbitrageurs also confront statistical obstacles (like "complexity" here) in learning the DGP of alphas. Da *et al.* (2022) show that regardless of the machine learning methods arbitrageurs use, they cannot realize the optimal infeasible Sharpe ratio in certain low signal-to-noise ratio environments. Moreover, even if arbitrageurs adopt an *optimal feasible* trading strategy, there remains a substantial wedge between their Sharpe ratio and the optimal infeasible one. We discuss more details of each of these papers in Chapter 4.6.

3

Return Prediction

Empirical asset pricing is about measuring asset risk premia. This measurement comes in two forms: one seeking to describe and understand differences in risk premia across assets, the other focusing on time series dynamics of risk premia. As the first moment of returns, it is a natural starting point for surveying the empirical literature of financial machine learning. The first moment at once summarizes i) the extent of discounting that investors settle on as fair compensation for holding risky assets (potentially distorted by frictions that introduce an element of mispricing), and ii) the first-order aspect of the investment opportunities available to a prospective investor.

A return prediction is, by definition, a measurement of an asset's conditional expected excess return:[1]

$$R_{i,t+1} = \mathrm{E}_t[R_{i,t+1}] + \epsilon_{i,t+1}, \tag{3.1}$$

Related to equation (1.2), $\mathrm{E}_t[R_{i,t+1}] = \mathrm{E}[R_{i,t+1}|\mathcal{I}_t]$ represents a forecast conditional on the information set $\mathcal{I}_t$, and $\epsilon_{i,t+1}$, collects all the remaining unpredictable variation in returns. When building statistical models of market data, it is worthwhile to step back and recognize that the

[1] $R_{i,t+1}$ is an asset's return in excess of the risk-free rate with assets indexed by $i = 1, ..., N_t$ and dates by $t = 1, ..., T$.

data is generated from an extraordinarily complicated process. A large number of investors who vary widely in their preferences and individual information sets interact and exchange securities to maximize their well-being. The traders intermittently settle on prices, and the sequence of price changes (and occasional cash flows such as stock dividends or bond coupons) produce a sequence of returns. The information set implied by the conditional expectation in (3.1) reflects all information—public, private, obvious, subtle, unambiguous, or dubious—that in some way influences prices decided at time t. We emphasize the complicated genesis of this conditional expectation because we next make the quantum leap of defining a concrete function to describe its behavior:

$$\mathrm{E}_t[R_{i,t+1}] = g^\star(z_{i,t}). \tag{3.2}$$

Our objective is to represent $\mathrm{E}_t[R_{i,t+1}]$ as an immutable but otherwise general function $g^\star$ of the P-dimensional predictor variables $z_{i,t}$ available to we the researchers. That is, we hope to isolate a function that, once we condition on $z_{i,t}$, fully explains the heterogeneity in expected returns across all assets and over all time, as $g^\star$ depends neither on i nor t.[2] By maintaining the same form over time and across assets, estimation can leverage information from the entire panel, which lends stability to estimates of expected returns for any individual asset. This is in contrast to some standard asset pricing approaches that re-estimate a cross-sectional model each time period, or that independently estimate time series models for each asset (see Giglio *et al.*, 2022a). In light of the complicated trading process that generates return data, this "universality" assumption is ambitious. First, it is difficult to imagine that researchers can condition on the same information set as market participants (the classic Hansen-Richard critique). Second, given the constantly evolving technological and cultural landscape surrounding markets, not to mention the human whims that can influence prices, the concept of a universal $g^\star(\cdot)$ seems far-fetched. So, while some economists might view this framework as excessively flexible (given that it allows for an arbitrary function and predictor set), it seems that (3.2) places

[2] Also, $g^\star(\cdot)$ depends on z only through $z_{i,t}$. In most of our analysis, predictions will not use information from the history prior to t, or from individual stocks other than the i^{th}, though this is generalized in some analyses that we reference later.

onerous restrictions on the model of expected returns. For those that view it as implausibly restrictive, then any ability of this framework to robustly describe various behaviors of asset returns—across assets, time, and particular on an out-of-sample basis—can be viewed as an improbable success.

Equations (3.1) and (3.2) offer an organizing template for the machine learning literature on return prediction. This section is organized by the functional form $g(z_{i,t})$ used in each paper to approximate the true prediction function $g^{\star}(z_{i,t})$ in our template. We emphasize the main empirical findings of each paper, and highlight new methodological contributions and distinguishing empirical results of individual papers. We avoid discussing detailed differences in conditioning variables in different papers, but the reader should be aware that these variations are also responsible for variation in empirical results (above and beyond differences in functional form).

There are two distinct strands of literature associated with the time series and cross section research agendas discussed above. The literature on time series machine learning models for aggregate assets (e.g., equity or bond index portfolios) developed earlier but is the smaller literature. Efforts in time series return prediction took off in the 1980's following Shiller (1981)'s documentation of the excess volatility puzzle. As researchers set out to quantify the extent of excess volatility in markets, there emerged an econometric pursuit of time-varying discount rates through predictive regression. Development of the time series prediction literature occurred earlier than the cross section literature due to the earlier availability of data on aggregate portfolio returns and due to the comparative simplicity of forecasting a single time series. The small size of the market return data set is also a reason for the limited machine learning analysis of this data. With so few observations (several hundred monthly returns), after a few research attempts one exhausts the plausibility that "out-of-sample" inferences are truly out-of-sample.

The literature on machine learning methods for predicting returns in panels of many individual assets is newer, much larger, and continues to grow. It falls under the rubric of "the cross section of returns" because early studies of single name stocks sought to explain differences in unconditional average returns across stocks—i.e., the data boiled down a single cross section of average returns. In its modern incarnation, however, the so-called "cross section" research program takes the form of a true panel prediction problem. The objective is to explain both time-variation and cross-sectional differences in conditional expected returns. A key reason for the rapid and continual expansion of this literature is the richness of the data. The panel's cross-section dimension can multiply the number of time series observations by a factor of a few thousand or more (in the case of single name stocks, bonds, or options, for example). While there is notable cross-correlation in these observations, most of the variation in single name asset returns is idiosyncratic. Furthermore, patterns in returns appear to be highly heterogeneous. In other words, the panel aspect of the problem introduces a wealth of phenomena for empirical researchers to explore, document, and attempt to understand in an economic frame.

We have chosen an organizational scheme for this section that categorizes the literature by machine learning methods employed, and in each section we discuss both time series and panel applications. Given the prevalence of cross section studies, this topic receives the bulk of our attention.

3.1 Data

Much of the financial machine learning literature studies a canonical data set consisting of monthly returns of US stocks and accompanying stock-level signals constructed primarily from CRSP-Compustat data. Until recently, there were few standardized choices for building this stock-month panel. Different researchers use different sets of stock-level predictors (e.g., Lewellen (2015) uses 15 signals, Freyberger *et al.* (2020) use 36, Gu *et al.* (2020b) use 94) and impose different observation filters (e.g., excluding observations stocks with nominal share prices below $5, excluding certain industries like financials or utilities, and so forth). And it is common for different papers to use different constructions for signals with the same name and economic rationale (e.g., various formulations of "value," "quality," and so forth).

Fortunately, recent research has made progress streamlining these data decisions by publicly releasing data and code for a standardized stock-level panel, available directly from the Wharton Research Data Services server. Jensen *et al.* (2021) construct 153 stock signals, and provide source code and documentation so that users can easily inspect, analyze, and modify empirical choices. Furthermore, their data is available not just for US stocks, but for stocks in 93 countries around the world, and is updated regularly to reflect annual CRSP-Compustat data releases. These resources may be accessed at jkpfactors.com. Jensen *et al.* (2021) emphasize a homogenous approach to signal construction by attempting to make consistent decisions in how different CRSP-Compustat data items are used in various signals. Chen and Zimmermann (2021) also post code and data for US stocks at openassetpricing.com.

In terms of standardized data for aggregate market return prediction, Welch and Goyal (2008) post updated monthly and quarterly returns and predictor variables for the aggregate US stock market.[3] Rapach and Zhou (2022) provide the latest review of additional market predictors.

3.2 Experimental Design

The process of estimating and selecting among many models is central to the machine learning definition given above. Naturally, selecting a best performing model according to in-sample (or training sample) fit exaggerates model performance since increasing model parameterization mechanically improves in-sample fit. Sufficiently large models will fit the training data exactly. Once model selection becomes part of the research process, we can no longer rely on in-sample performance to evaluate models.

Common approaches for model selection are based on information criteria or cross-validation. An information criterion like Akaike (AIC) or Bayes/Schwarz (BIC) allows researchers to select among models based on training sample performance by introducing a probability-theoretical performance penalty related to the number of model parameters. This serves as a counterbalance to the mechanical improvement in fit due to heavier parameterization. Information criteria aim to select a model from the candidate set that is likely to have the best out-of-sample

[3] Available at sites.google.com/view/agoyal145.

prediction performance according to some metric.

Cross-validation has the same goal as AIC and BIC, but approaches the problem in a more data-driven way. It compares models based on their "pseudo"-out-of-sample performance. Cross-validation separates the observations into one set used for training and another (the pseudo-out-of-sample observations) for performance evaluation. By separating the training sample from the evaluation sample, cross-validation avoids the mechanical outperformance of larger models by simulating out-of-sample model performance. Cross-validation selects models based on their predictive performance in the pseudo-out-of-sample data.

Information criteria and cross-validation each have their advantages and disadvantages. In some circumstances, AIC and cross-validation deliver asymptotically equivalent model selections (Stone, 1977; Nishii, 1984). A disadvantage of information criteria is that they are derived from certain theoretical assumptions, so if the data or models violate these assumptions, the theoretical criteria must be amended and this may be difficult or infeasible. Cross-validation implementations are often more easily adapted to account for challenging data properties like serial dependence or extreme values, and can be applied to almost any machine learning algorithm (Arlot and Celisse, 2010). On the other hand, due to its random and repetitive nature, cross-validation can produce noisier model evaluations and can be computationally expensive. Over time, the machine learning literature has migrated toward a heavy reliance on cross-validation because of its apparent adaptivity and universality, and thanks to the reduced cost in high-performance computing.

To provide a concrete perspective on model selection with cross-validation and how it fits more generally into machine learning empirical designs, we outline example designs adopted by Gu *et al.* (2020b) and a number of subsequent studies.

3.2.1 Example: Fixed Design

Let the full research sample consist of time series observations indexed $t = 1, ..., T$. We begin with an example of a fixed sample splitting scheme. The full sample of T observations is split into three disjoint subsamples. The first is the "training" sample which is used to estimate all candidate models. The training sample includes the T_{train} observations from $t = 1, ..., T_{\text{train}}$. The set of candidate models is often governed by a set of "hyperparameters" (also commonly called "tuning parameters"). An example of a hyperparameter that defines a continuum of candidate models is the shrinkage parameter in a ridge regression, while an example of a tuning parameter that describes a discrete set of models is the number of components selected from PCA.

The second, or "validation," sample consists of the pseudo-out-of-sample observations. Forecasts for data points in the validation sample are constructed based on the estimated models from the training sample. Model performance (usually defined in terms of the objective function of the model estimator) on the validation sample determines which specific hyperparameter values (i.e., which specific models) are selected. The validation sample includes the T_{validate} observations $t = T_{\text{train}} + 1, ..., T_{\text{train}} + T_{\text{validate}}$. Note that while the original model is estimated from only the first T_{train} data points, once a model specification is selected its parameters are typically re-estimated using the full $T_{\text{train}} + T_{\text{validate}}$ observations to exploit the full efficiency of the in-sample data when constructing out-of-sample forecasts.

The validation sample fits are of course not truly out-of-sample because they are used for tuning, so validation performance is itself subject to selection bias. Thus a third "testing" sample (used for neither estimation nor tuning) is used for a final evaluation of a method's predictive performance. The testing sample includes the T_{test} observations $t = T_{\text{train}} + T_{\text{validate}} + 1, ..., T_{\text{train}} + T_{\text{validate}} + T_{\text{test}}$.

Two points are worth highlighting in this simplified design example. First, the final model in this example is estimated once and for all using data through $T_{\text{train}} + T_{\text{validate}}$. But, if the model is re-estimated recursively throughout the test period, the researcher can produce more efficient out-of-sample forecasts. A reason for relying on a fixed sample split would be that the candidate models are very computationally intensive to train, so re-training them may be infeasible or incur large computing costs (see, e.g., the CNN analysis of Jiang *et al.*, 2022).

Second, the sample splits in this example respect the time series ordering of the data. The motivation for this design is to avoid inadvertent information leakage backward in time. Taking this further, it

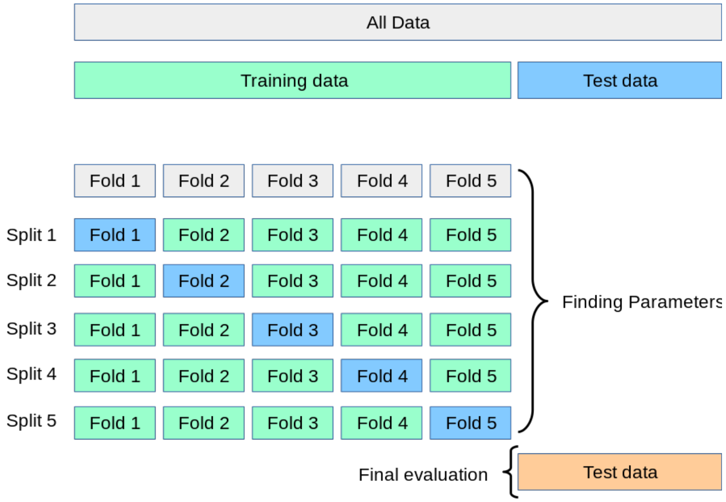


Figure 3.1: Illustration of Standard K-fold Cross-validation

Source: https://scikit-learn.org/

is common in time series applications to introduce an embargo sample between the training and validation samples so that serial correlation across samples does not bias validation. For stronger serial correlation, longer embargoes are appropriate.

Temporal ordering of the training and validation samples is not strictly necessary and may make inefficient use of data for the model selection decision. For example, a variation on the temporal ordering in this design would replace the fixed validation sample with a more traditional K-fold cross-validation scheme. In this case, the first $T_{\text{train}} + T_{\text{validate}}$ observations can be used to produce K different validation samples, from which a potentially more informed selection can be made. This would be appropriate, for example, with serially uncorrelated data. Figure 3.1 illustrates the K-fold cross-validation scheme, which creates K validation samples over which performance is averaged to make a model selection decision.

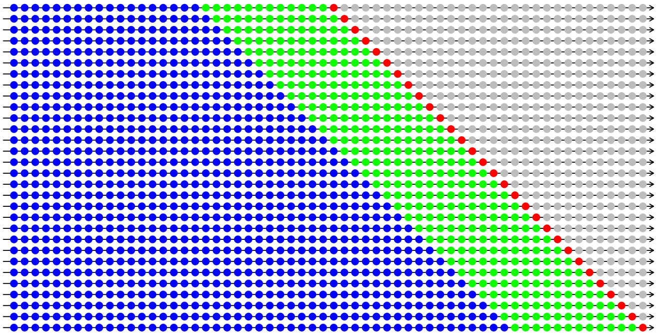


Figure 3.2: Illustration of Recursive Time-ordered Cross-validation
Note: Blue dots represent training observations, green dots represent validation observations, and red dots represent test observations. Each row represents a step in the recursive design. This illustration corresponds to the case of an expanding (rather than rolling) training window.

3.2.2 Example: Recursive Design

When constructing an out-of-sample forecast for a return realized at some time t, an analyst typically wishes to use the most up-to-date sample to estimate a model and make out-of-sample forecasts. In this case, the training and validation samples are based on observations at $1, ..., t-1$. For example, for a 50/50 split into training/validation samples, the fixed design above would be adapted to train on $1, ..., \lfloor \frac{t-1}{2} \rfloor$ and validate on $\lfloor \frac{t-1}{2} \rfloor + 1, ..., t-1$. Then the selected model is re-estimated using all data through $t-1$ and an out-of-sample forecast for t is generated.

At $t+1$, the entire training/validation/testing processes is repeated again. Training uses observations $1, ..., \lfloor \frac{t}{2} \rfloor$, validation $\lfloor \frac{t}{2} \rfloor + 1, ..., t$, and the selected model is re-estimated through t to produce out-of-sample forecast. This recursion iterates until a last out-of-sample forecast is generated for observation T. Note that, because validation is re-conducted each period, the selected model can change throughout the recursion. Figure 3.2 illustrates this recursive cross-validation scheme.

A common variation on this design is to use rolling a training window rather than an expanding window. This is beneficial if there is suspicion of structural instability in the data or if there are other modeling or testing benefits to maintaining equal training sample sizes throughout the recursion.

3.3 A Benchmark: Simple Linear Models

The foundational panel model for stock returns, against which any machine learning method should be compared, is the simple linear model. For a given set of stock-level predictive features $z_{i,t}$, the linear panel model fixes the prediction function as $g(z_{i,t}) = \beta' z_{i,t}$:

$$R_{i,t+1} = \beta' z_{i,t} + \epsilon_{i,t+1}. \tag{3.3}$$

There are a variety of estimators for this model that are appropriate under various assumptions on structure of the error covariance matrix. In empirical finance research, the most popular is Fama and Macbeth (1973) regression. Petersen (2008) analyzes the econometric properties of Fama-MacBeth and compare it with other panel estimators.

Haugen and Baker (1996) and Lewellen (2015) are precursors to the literature on machine learning for the cross section of returns. First, they employ a comparatively large number of signals: Haugen and Baker (1996) use roughly 40 continuous variables and sector dummies, and Lewellen (2015) uses 15 continuous variables. Second, they recursively train the panel linear model in (3.3) and emphasize the out-of-sample performance of their trained models. This differentiates their analysis from a common empirical program in the literature that sorts stocks into bins on the basis of one or two characteristics—which is essentially a "zero-parameter" return prediction model that asserts a functional form for $g(z_{i,t})$ without performing estimation.[4] Both Haugen and Baker (1996) and Lewellen (2015) evaluate out-of-sample model performance in the economic terms of trading strategy performance. Haugen and Baker (1996) additionally analyze international data, while Lewellen (2015) additionally analyzes model accuracy in terms of prediction R^2.

[4] To the best of our knowledge Basu (1977) is the first to perform a characteristic-based portfolio sort. He performs quintile sorts on stock-level price-earnings ratios. The adoption of this approach to by Fama and French (1992) establishes portfolio sorts as a mainstream methodology for analyzing the efficacy of a candidate stock return predictor.

Table 3.1: Average Monthly Factor Returns From Haugen and Baker (1996)

	1979/01 through 1986/06		1986/07 through 1993/12	
Factor	Mean	t-stat	Mean	t-stat
One-month excess return	-0.97%	-17.04	-0.72%	-11.04
Twelve-month excess return	0.52%	7.09	0.52%	7.09
Trading volume/market cap	-0.35%	-5.28	-0.20%	-2.33
Two-month excess return	-0.20%	-4.97	-0.11%	-2.37
Earnings to price	0.27%	4.56	0.26%	4.42
Return on equity	0.24%	4.34	0.13%	2.06
Book to price	0.35%	3.90	0.39%	6.72
Trading volume trend	-0.10%	-3.17	-0.09%	-2.58
Six-month excess return	0.24%	3.01	0.19%	2.55
Cash flow to price	0.13%	2.64	0.26%	4.42
Variability in cash flow to price	-0.11%	-2.55	-0.15%	-3.38

Source: Haugen and Baker (1996).

Haugen and Baker (1996) show that optimized portfolios built from the linear model's return forecasts outperform the aggregate market, and does so in each of the five countries they study. One of their most interesting findings is the stability in predictive patterns, recreated in Table 3.1 (based on Table 1 in their paper). Coefficients on the important predictors in the first half of their sample have not only the same sign but strikingly similar magnitudes and t-statistics in the second half of their sample.

Lewellen (2015) shows that, in the cross section of all US stocks, the panel linear model has a highly significant out-of-sample R^2 of roughly 1% per month, demonstrating its capability of quantitatively aligning the *level* of return forecasts with realized returns. This differentiates the linear model from the sorting approach, which is usually evaluated on its ability to significantly distinguish high and low expected return stocks—i.e., its ability to make relative comparisons without necessarily matching magnitudes. Furthermore, the 15-variable linear model translates into impressive trading strategy performance, as shown in Table 3.2 (recreated from Table 6A of Lewellen (2015)). An equal-weight long-short strategy that buys the highest model-based expected return decile and shorts the lowest earns an annualized Sharpe ratio of 1.72 on an out-of-sample basis (0.82 for value-weight deciles). Collectively, the

Table 3.2: Average Monthly Factor Returns And Annualized Sharpe Ratios From Lewellen (2015)

	Equal-weighted					Value-weighted				
	Pred	Avg	Std	t-stat	Shp	Pred	Avg	Std	t-stat	Shp
Panel A: All stocks										
Low (L)	-0.90	-0.32	7.19	-0.84	-0.15	-0.76	0.11	6.01	0.37	0.06
2	-0.11	0.40	5.84	1.30	0.24	-0.10	0.45	4.77	1.89	0.32
3	0.21	0.60	5.46	2.06	0.38	0.21	0.65	4.65	2.84	0.49
4	0.44	0.78	5.28	2.74	0.51	0.44	0.69	4.67	2.97	0.51
5	0.64	0.81	5.36	2.82	0.52	0.63	0.81	5.01	3.34	0.56
6	0.83	1.04	5.36	3.62	0.67	0.82	0.88	5.22	3.28	0.58
7	1.02	1.12	5.55	3.68	0.70	1.01	1.04	5.67	3.46	0.64
8	1.25	1.31	5.97	4.04	0.76	1.24	1.15	6.03	3.62	0.66
9	1.55	1.66	6.76	4.38	0.85	1.54	1.34	6.68	3.80	0.69
High (H)	2.29	2.17	7.97	4.82	0.94	2.19	1.66	8.28	3.73	0.70
H - L	3.20	2.49	5.02	10.00	1.72	2.94	1.55	6.56	4.51	0.82

Source: Lewellen (2015).

evidence of Haugen and Baker (1996) and Lewellen (2015) demonstrates that simple linear panel models can, in real time, estimate combinations of many predictors that are effective for forecasting returns and building trading strategies.

Moving to time series analysis, the excess volatility puzzle of Shiller (1981) prompted a large literature seeking to quantify the extent of time-variation in discount rates, as well as a productive line of theoretical work rationalizing the dynamic behavior of discount rates (e.g. Campbell and Cochrane, 1999; Bansal and Yaron, 2004; Gabaix, 2012; Wachter, 2013). The tool of choice in the empirical pursuit of discount rate variation is linear time series regression. As noted in the Rapach and Zhou (2013) survey of stock market predictability, the most popular predictor is the aggregate price-dividend ratio (e.g. Campbell and Shiller, 1988), though dozens of other predictors have been studied. By and large, the literature focuses on univariate or small multivariate prediction models, occasionally coupled with economic restrictions such as non-negativity constraints on the market return forecast (Campbell and Thompson, 2008) and imposing cross-equation restrictions in the present-value identity (Cochrane, 2008; Van Binsbergen and Koijen, 2010). In an influential critique, Welch and Goyal (2008) contend that the abundant in-sample evidence of market return predictability from simple linear models fails to generalize out-of-sample. However, Rapach *et al.* (2010) show that forecast combination techniques produce reliable out-of-sample market forecasts.

3.4 Penalized Linear Models

To borrow a phrase from evolutionary biology, the linear models of Haugen and Baker (1996) and Lewellen (2015) are a "transitional species" in the finance literature. Like traditional econometric approaches, the researchers fix their model specifications a priori. But like machine learning, they consider a much larger set of predictors than their predecessors and emphasize out-of-sample forecast performance.

While these papers study dozens of return predictors, the list of predictive features analyzed in the literature numbers in the hundreds (Harvey *et al.*, 2016; Hou *et al.*, 2018; Jensen *et al.*, 2021). Add to this an interest in expanding the model to incorporate state-dependence in predictive relationships (e.g. Schaller and Norden, 1997; Cujean and Hasler, 2017), and the size of a linear model's parameterization quickly balloons to thousands of parameters. Gu *et al.* (2020b) consider a baseline linear specification to predict the panel of stock-month returns. They use approximately 1,000 predictors that are multiplicative interactions of roughly 100 stock characteristics with demonstrated forecast power for individual stock returns and 10 aggregate macro-finance predictors with demonstrated success in predicting the market return. Despite the fact that these predictors have individually shown promise in prior research, Gu *et al.* (2020b) show that OLS cannot achieve a stable fit of a model with so many parameters at once, resulting in disastrous out-of-sample performance. The predictive R^2 is -35% per month and a trading strategy based on these predictions underperforms the market.

It is hardly surprising that OLS estimates fail with so many predictors. When the number of predictors P approaches the number of observations NT, the linear model becomes inefficient or even inconsistent. It begins to overfit noise rather than extracting signal. This is particularly troublesome for the problem of return prediction where the signal-to-noise ratio is notoriously low.

A central conclusion from the discussion of such "complex" models in Section 2 is the following: Crucial for avoiding overfit is constraining the model by regularizing the estimator. This can be done by pushing complexity (defined as c in Section 2) far above one (which implicitly regularizes the least squares estimator) or by imposing explicit penalization via ridge or other shrinkage. The simple linear model does neither. Its complexity is in an uncomfortably high variance zone (well above zero, but not above one) and it uses no explicit regularization.

The prediction function for the penalized linear model is the same as the simple linear model in equation (3.3). That is, it continues to consider only the baseline, untransformed predictors. Penalized methods differ by appending a penalty to the original loss function, such as the popular "elastic net" penalty, resulting in a penalized loss of

$$\mathcal{L}(\beta;\rho,\lambda)=\sum_{i=1}^{N}\sum_{t=1}^{T}\left(R_{i,t+1}-\beta' z_{i,t}\right)^2+\lambda(1-\rho)\sum_{j=1}^{P}|\beta_j|+\frac{1}{2}\lambda\rho\sum_{j=1}^{P}\beta_j^2 \tag{3.4}$$

The elastic net involves two non-negative hyperparameters, λ and ρ, and includes two well known regularizers as special cases. The $\rho = 1$ case corresponds to ridge regression, which uses an ℓ_2 parameter penalization, that draws all coefficient estimates closer to zero but does not impose exact zeros anywhere. Ridge is a shrinkage method that helps prevent coefficients from becoming unduly large in magnitude. The $\rho = 0$ case corresponds to the lasso and uses an absolute value, or "ℓ_1", parameter penalization. The geometry of lasso sets coefficients on a subset of covariates to exactly zero if the penalty is large enough. In this sense, the lasso imposes sparsity on the specification and can thus be thought of as both a variable selection and a shrinkage device. For intermediate values of ρ, the elastic net encourages simple models with varying combinations of ridge and lasso effects.

Gu *et al.* (2020b) show that the failure of the OLS estimator in the stock-month return prediction panel is reversed by introducing elastic net penalization. The out-of-sample prediction R^2 becomes positive and an equal-weight long-short decile spread based on elastic net predictions returns an out-of-sample Sharpe ratio of 1.33 per annum. In other words, it is not the weakness of the predictive information embodied in the 1,000 predictors, but the statistical cost (overfit and inefficiency) of the heavy parameter burden, that is a detriment to the performance of OLS. He *et al.* (2022a) analyze elastic-net ensembles and find that the best individual elastic-net models change rather quickly over time and that ensembles perform well at capturing these changes. Rapach *et al.* (2013) apply (3.4) to forecast market returns across countries.

Penalized regression methods are some of the most frequently employed machine learning tools in finance, thanks in large part to their conceptual and computational tractability. For example, the ridge regression estimator is available in closed form so it has the same computational simplicity as OLS. There is no general closed form representation of the lasso estimator, so it must be calculated numerically, but efficient algorithms for computing lasso are ubiquitous in statistics software packages (including Stata, Matlab, and various Python libraries).

Freyberger *et al.* (2020) combine penalized regression with a generalized additive model (GAM) to predict the stock-month return panel. In their application, a function $p_k(z_{i,t})$ is an element-wise nonlinear transformation of the P variables in $z_{i,t}$:

$$g(z_{i,t}) = \sum_{k=1}^{K} \tilde{\beta}_k' p_k(z_{i,t}). \tag{3.5}$$

Their model expands the set of predictors by using $k = 1, ..., K$ such nonlinear transformations, with each transformation k having its own $Q \times 1$ vector of linear regression coefficients $\tilde{\beta}_k$. Freyberger *et al.* (2020) use a quadratic spline formulation for their basis functions, though the basis function possibilities are endless and the nonlinear transformations may be applied to multiple predictors jointly (as opposed to element-wise).

Because nonlinear terms enter additively, forecasting with the GAM can be approached with the same estimation tools as any linear model. But the series expansion concept that underlies the GAM quickly multiplies the number of model parameters, thus penalization to control the degrees of freedom tends to benefit out-of-sample performance. Freyberger *et al.* (2020) apply a penalty function known as group lasso

(Huang *et al.*, 2010), which takes the form

$$\lambda \sum_{j=1}^{Q} \left(\sum_{k=1}^{K} \tilde{\beta}_{k,j}^{2} \right)^{1/2}, \tag{3.6}$$

where $\tilde{\beta}_{k,j}$ is the coefficient on the k^{th} basis function applied to stock signal j. This penalty is particularly well suited to the spline expansion setting. As its name suggests, group lasso selects either all K spline terms associated with a given characteristic j, or none of them.

Freyberger *et al.* (2020)'s group lasso results show that less than half of the commonly studied stock signals in the literature have independent predictive power for returns. They also document the importance of nonlinearities, showing that their full nonlinear specification dominates the nested linear specification in terms of out-of-sample trading strategy performance.

Chinco *et al.* (2019) also use lasso to study return predictability. Their study has a number of unique aspects. First is their use of high frequency data—they forecast one-minute-ahead stock returns trained in rolling 30-minute regressions. Second, they use completely separate models for each stock, making their analysis a large collection of time series linear regressions. And, rather than using standard stock-level characteristics as predictors, their feature set includes three lags of one-minute returns for *all* stocks in the NYSE cross section. Perhaps the most interesting aspect of this model is its accommodation of cross-stock prediction effects. Their ultimate regression specification is

$$R_{i,t} = \alpha_i + \beta_{i,1}' R_{t-1} + \beta_{i,2}' R_{t-2} + \beta_{i,3}' R_{t-3} + \epsilon_{i,t}, \quad i = 1, ..., N \tag{3.7}$$

where R_t is the vector including all stocks' returns in minute t. The authors estimate this model for each stock i with lasso using a stock-specific penalty parameter selected with 10-fold cross-validation. They find that the dominant predictors vary rather dramatically from period to period, and tend to be returns on stocks that are reporting fundamental news. The economic insights from these patterns are not fully fleshed out, but the strength of the predictive evidence suggest that machine learning methods applied to high frequency returns have the potential to reveal new and interesting phenomena relating to information flows and the joint dynamics they induce among assets.

Avramov *et al.* (2022b) study how dynamics of firm-level fundamentals associate with subsequent drift in a firm's stock price. They take an agnostic view on details of fundamental dynamics. Their data-driven approach considers the deviations of all quarterly Compustat data items from their mean over the three most recent quarters, rather than hand-picking specific fundamentals a priori. This large set of deviations is aggregated into a single return prediction index via supervised learning. In particular, they estimate pooled panel lasso regressions to forecast the return on stock i using all of stock i's Compustat deviations, and refer to the fitted value from these regressions as the Fundamental Deviation Index, or FDI. A value-weight decile spread strategy that is long highest FDI stocks and short lowest FDI stocks earns an annualized out-of-sample information ratio of 0.8 relative to the Fama-French-Carhart four-factor model.

3.5 Dimension Reduction

The regularization aspect of machine learning is in general beneficial for high dimensional prediction problems because it reduces degrees of freedom. There are many possible ways to achieve this. Penalized linear models reduce degrees of freedom by shrinking coefficients toward zero and/or forcing coefficients to zero on a subset of predictors. But this can produce suboptimal forecasts when predictors are highly correlated. Imagine a setting in which each predictor is equal to the forecast target plus some i.i.d. noise term. In this situation, the sensible forecasting solution is to simply use the average of predictors in a univariate predictive regression.

This idea of predictor averaging is the essence of dimension reduction. Forming linear combinations of predictors helps reduce noise to better isolate signal. We first discuss two classic dimension reduction techniques, principal components regression (PCR) and partial least squares (PLS), followed by two extensions of PCA, scaled PCA and supervised PCA, designed for low signal-to-noise settings. These methods emerge in the literature as dimension-reduction devices for time series prediction of market returns or macroeconomic variables. We next extend their use to a panel setting for predicting the cross-section of returns, and then introduce a more recent finance-centric method known as principal portfolios analysis tailored to this problem. In this section, we focus on applications of dimension reduction to prediction. Dimension reduction plays an important role in asset pricing beyond prediction, and we study these in subsequent sections. For example, a variety of PCA-based methods are at the heart of latent factor analysis, and are treated under the umbrella of machine learning factor pricing models in Section 4.

3.5.1 Principal Components and Partial Least Squares

We formalize the discussion of these two methods in a generic predictive regression setting:

$$y_{t+h} = x_t'\theta + \epsilon_{t+h}, \tag{3.8}$$

where y may refer to market return or macroeconomic variables such as GDP growth, unemployment, and inflation, x_t is $P \times 1$ vector of predictors, and h is the prediction horizon.

The idea of dimension reduction is to replace the high-dimensional predictors by a set of low-dimensional "factors", f_t, which summarizes useful information in x_t. Their relationship is often cast in a standard factor model:

$$x_t = \beta f_t + u_t. \tag{3.9}$$

On the basis of (3.9), we can rewrite (3.8) as:

$$y_{t+h} = f_t'\alpha + \tilde{\epsilon}_{t+h}. \tag{3.10}$$

Equations (3.9) and (3.10) can be represented in matrix form as:

$$X = \beta F + U, \quad \overline{Y} = \underline{F}\alpha + \overline{E}, \tag{3.11}$$

where X is a $P \times T$ matrix, F is $K \times T$, $\underline{F}$ is the $K \times (T-h)$ matrix with the last h columns removed from F, $\overline{Y} = (y_{h+1}, y_{h+2}, \ldots, y_T)'$, $\overline{E} = (\tilde{\epsilon}_{h+1}, \tilde{\epsilon}_{h+2}, \ldots, \tilde{\epsilon}_T)'$.

Principal components regression (PCR) is a two-step procedure. First, it combines predictors into a few linear combinations, $\widehat{f}_t = \Omega_K x_t$, and finds the combination weights Ω_K recursively. The j^{th} linear combination solves

$$\begin{aligned} w_j &= \arg\max_{w} \widehat{\mathrm{Var}}(x_t' w), \\ &\text{s.t.} \quad w'w = 1, \quad \widehat{\mathrm{Cov}}(x_t' w, x_t' w_l) = 0, \quad l = 1, 2, \ldots, j-1. \end{aligned} \tag{3.12}$$

Clearly, the choice of components is not based on the forecasting objective at all, but aims to best preserve the covariance structure among the predictors.[5] In the second step, PCR uses the estimated components $\widehat{f}_t$ in a standard linear predictive regression (3.10).

Stock and Watson (2002) propose to forecast macroeconomic variables on the basis of PCR. They prove the consistency of the first-stage recovery of factors (up to a rotation) and the second-stage prediction, as both the number of predictors, P, and the sample size, T, increase.

Among the earliest uses of PC forecasts for stock return prediction is Ludvigson and Ng (2007). Their forecast targets are either the quarterly return to the CRSP value-weighted index or its quarterly volatility. They consider two sets of principal components, corresponding to different choices for the raw predictors that comprise X. The first is a large collection of indicators spanning all aspects of the macroeconomy (output, employment, housing, price levels, and so forth). The second is a large collection of financial data, consisting mostly of aggregate US market price and dividend data, government bond and credit yields, and returns on various industry and characteristic-sorted portfolios. In total, they incorporate 381 individual predictors in their analysis. They use BIC to select model specifications that include various combinations of the estimated components. Components extracted from financial predictors have significant out-of-sample forecasting power for market returns and volatility, while macroeconomic indicators do not. Fitted mean and volatility forecasts exhibit a positive association providing evidence of a classical risk-return tradeoff at the aggregate market level. To help interpret their predictions, the authors show that *"Two factors stand out as particularly important for quarterly excess returns: a volatility factor that is highly correlated with squared returns,*

[5] This optimization problem is efficiently solved via singular value decomposition of X, a $P \times T$ matrix with columns in $\{x_1, \ldots, x_T\}$.

and a risk-premium factor that is highly correlated with well-established risk factors for explaining the cross-section of expected returns."

Extending their earlier equity market framework, Ludvigson and Ng (2010) use principal components of macroeconomic and financial predictors to forecast excess Treasury bond returns. They document reliable bond return prediction performance in excess of that due to forward rates (Fama and Bliss, 1987; Cochrane and Piazzesi, 2005). They emphasize i) the inconsistency of this result with leading affine term structure models (in which forward rates theoretically span all predictable variation in future bond returns) and ii) that their estimated conditional expected returns are decidedly countercyclical, resolving the puzzling cyclicality of bond risk premia that arises when macroeconomic components are excluded from the forecasting model.

Jurado *et al.* (2015) use a clever application of PCR to estimate macroeconomic risk. The premise of their argument is that a good conditional variance measure must effectively adjust for conditional means,

$$\mathrm{Var}(y_{t+1}|\mathcal{I}_t) = \mathrm{E}\left[\left(y_{t+1} - \mathrm{E}[y_{t+1}|\mathcal{I}_t]\right)^2 |\mathcal{I}_t\right].$$

If the amount of mean predictability is underestimated, conditional risks will be overestimated. The authors build on the ideas in Ludvigson and Ng (2007) and Ludvigson and Ng (2010) to saturate predictions of market returns (and other macroeconomic series) with information contained in the principal components of macroeconomic predictor variables. The resulting improvements in macroeconomic risk estimates have interesting economic consequences. First, episodes of elevated uncertainty are fewer and farther between than previously believed. Second, improved estimates reveal a tighter link between rises in risk and depressed macroeconomic activity.

PCR constructs predictors solely based on covariation among the predictors. The idea is to accommodate as much of the total variation among predictors as possible using a relatively small number of dimensions. This happens prior to and independent of the forecasting step. This naturally suggests a potential shortcoming of PCR—that it fails to consider the ultimate forecasting objective in how it conducts dimension reduction.

Partial least squares (PLS) is an alternative to PCR that reduces dimensionality by directly exploiting covariation between predictors and the forecast target. Unlike PCR, PLS seeks components of X that maximize predictive correlation with the forecast target, thus weight in the j^{th} PLS component are

$$\begin{aligned} w_j = &\arg\max_{w} \ \widehat{\text{Cov}}(y_{t+h}, x_t'w)^2, \\ &\text{s.t.} \quad w'w = 1, \quad \widehat{\text{Cov}}(x_t'w, x_t'w_l) = 0, \quad l = 1, 2, \ldots, j-1. \end{aligned} \tag{3.13}$$

At the core of PLS is a collection of univariate ("partial") models that forecast y_{t+h} one predictor at a time. Then, it constructs a linear combination of predictors weighted by their univariate predictive ability. To form multiple PLS components, the target and all predictors are orthogonalized with respect to previously constructed components, and the procedure is repeated on the orthogonalized dataset.

Kelly and Pruitt (2015) analyze the econometric properties of PLS prediction models (and a generalization called the three-pass regression filter), and note its resilience when the predictor set contains dominant factors that are irrelevant for prediction, which is a situation that limits the effectiveness of PCR. Related to this, Kelly and Pruitt (2013) analyze the predictability of aggregate market returns using the present-value identity. They note that traditional present-value regressions (e.g. Campbell and Shiller, 1988) face an errors-in-variables problem, and propose a high-dimensional regression solution that exploits strengths of PLS. In their analysis, Z consists of valuation (book-to-market or dividend-to-price) ratios for a large cross section of assets. Imposing economic restrictions, they derive a present-value system that relates market returns to the cross section of asset-level valuation ratios. However, the predictors are also driven by common factors in expected aggregate dividend growth. Applying PCR to the cross section of valuation ratios encounters the problem that factors driving dividend growth are not particularly useful for forecasting market returns. The PLS estimator learns to bypass these high variance but low predictability components in favor of components with stronger return predictability. Their PLS-based forecasts achieve an out-of-sample R^2 of 13% for annual returns. This translates into large economic gains for investors willing to time the market, increasing Sharpe ratios by more than a third relative to a buy-and-hold investor. Furthermore, they document substantially larger variability in investor discount rates than accommodated by leading theoretical models. Chatelais *et al.* (2023) use a similar framework to forecast macroeconomic activity using a cross section of asset prices, in essence performing a PLS-based version of the Fama (1990) analysis demonstrating that asset prices lead macroeconomic outcomes.

Baker and Wurgler (2006) and Baker and Wurgler (2007) use PCR to forecast market returns based on a collection of market sentiment indicators. Huang *et al.* (2014) extend this analysis with PLS and show that PLS sentiment indices possess significant prediction benefits relative to PCR. They argue that PLS avoids a common but irrelevant factor associated with measurement noise in the Baker-Wurgler sentiment proxies. By reducing the confounding effects of this noise, Huang *et al.* (2014) find that sentiment is a highly significant driver of expected returns, and that this predictability is in large part undetected by PCR. Similarly, Chen *et al.* (2022b) combine multiple investor attention proxies into a successful PLS-based market return forecaster.

Ahn and Bae (2022) conduct asymptotic analysis of the PLS estimator and find that the optimal number of PLS factors for forecasting could be much smaller than the number of common factors in the original predictor variables. Moreover, including too many PLS factors is detrimental to the out-of-sample performance of the PLS predictor.

3.5.2 Scaled PCA and Supervised PCA

A convenient and critical assumption in the analysis of PCR and PLS is that the factors are pervasive. Pervasive factor models are prevalent in the literature. For instance, Bai (2003) studies the asymptotic properties of the PCA estimator of factors in the case $\lambda_K(\beta'\beta) \gtrsim P$. Giglio *et al.* (2022b) show that the performance of the PCA and PLS predictors hinges on the signal-to-noise ratio, defined by $P/(T\lambda_K(\beta'\beta))$, where $\lambda_K(\beta'\beta)$ is the Kth largest eigenvalue. When $P/(T\lambda_K(\beta'\beta)) \not\to 0$, neither prediction method is consistent in general. This is known as the weak factor problem.[6]

Huang *et al.* (2021) propose a scaled-PCA procedure, which assigns weights to variables based on their correlations with the prediction target, before applying PCA. The weighting scheme enhances the signal-to-noise ratio and thus helps factor recovery.

Likewise, Giglio *et al.* (2022b) propose a supervised PCA (SPCA) alternative which allows for factors along a broad spectrum of factor strength. They note that the strength of factors depends on the set of variables to which PCA is applied. SPCA involves a marginal screening step to select a subset of predictors within which at least one factor is strong. It then extracts a first factor from the subset using PCA, projects the target and all the predictors (including those not selected) on the first factor, and constructs residuals. It then repeats the selection, PCA, and projection step with the residuals, extracting factors one by one until the correlations between residuals and the target vanish. Giglio *et al.* (2022b) prove that both PLS and SPCA can recover weak factors that are correlated with the prediction target and that the resulting predictor achieves consistency. In a multivariate target setting, if all factors are correlated with at least one of the prediction targets, the PLS procedure can recover the number of weak factors and the entire factor space consistently.

3.5.3 From Time-Series to Panel Prediction

The dimension reduction applications referenced above primarily focus on combining many predictors to forecast a univariate time series. To predict the cross-section of returns, we need generalize dimension reduction to a panel prediction setting as in (3.3). Similar to (3.9) and (3.10), we can write

$$R_{t+1} = F_t\alpha + E_{t+1}, \quad Z_t = F_t\gamma + U_t,$$

where R_{t+1} is an $N \times 1$ vector of returns observed on day $t+1$, F_t is an $N \times K$ matrix, α is $K \times 1$, E_{t+1} is $N \times 1$ vector of residuals, Z_t is an

[6] Bai and Ng (2021) extend their analysis to moderate factors, i.e., $P/(T\lambda_K(\beta'\beta)) \to 0$, and find PCA remains consistent. Earlier work on weak factor models includes Onatski (2009), Onatski (2010), and Onatski (2012), who consider the extremely weak factor setting in which the factors cannot be recovered consistently.

$N \times P$ matrix of characteristics, γ is $K \times P$, and U_t is $N \times P$ matrix of residuals.

We can then stack $\{R_{t+1}\}, \{E_{t+1}\}, \{Z_t\}, \{F_t\}, \{U_t\}$ into $NT \times 1$, $NT \times 1$, $NT \times P$, $NT \times K$, $NT \times P$ matrices, $\overline{R}$, $\overline{E}$, Z, F, and U, respectively, such that

$$\overline{R} = \underline{F}\alpha + \overline{E}, \quad Z = F\gamma + U.$$

These equations follow exactly the same form as (3.11), so that the aforementioned dimension reduction procedures apply.

Light *et al.* (2017) apply pooled panel PLS in stock return prediction, and Gu *et al.* (2020b) perform pooled panel PCA and PLS to predict individual stocks returns. The Gu *et al.* (2020b) equal-weighted long-short portfolios based on PCA and PLS earn a Sharp ratio of 1.89 and 1.47 per annum, respectively, which outperforms the elastic-net based long-short portfolio.

3.5.4 Principal Portfolios

Kelly *et al.* (2020a) propose a different dimension reduction approach to return prediction and portfolio optimization called "principal portfolios analysis" (PPA). In the frameworks of the preceding section, high dimensionality comes from each individual asset having many potential predictors. Most models in empirical asset pricing focus on own-asset predictive signals; i.e., the association between $S_{i,t}$ and the return on only asset i, $R_{i,t+1}$. PPA is motivated by a desire to harness the joint predictive information for many assets simultaneously. It leverages predictive signals of all assets to forecast returns on all other assets. In this case, the high dimensional nature of the problem comes from the large number of potential cross prediction relationships.

For simplicity, suppose we have a single signal, $S_{i,t}$, for each asset (stacked into an N-vector, S_t). PPA begins from the cross-covariance matrix of all assets' future returns with all assets' signals:

$$\Pi = \mathrm{E}(R_{t+1}S_t') \in \mathbb{R}^{N \times N}.$$

Kelly *et al.* (2020a) refer to Π as the "prediction matrix." The diagonal part of the prediction matrix tracks the own-signal prediction effects,

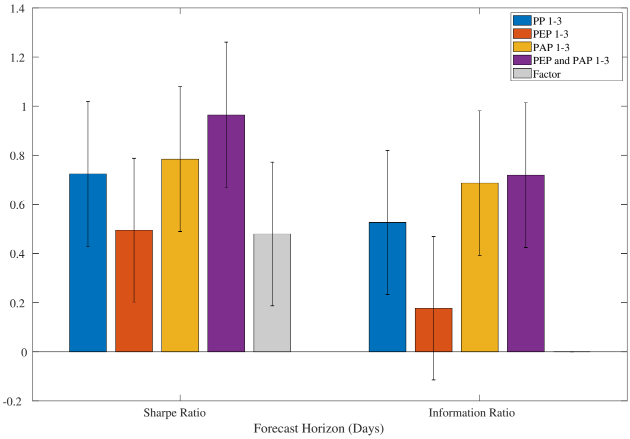


Figure 3.3: Principal portfolio performance ratios.

Source: Figure 3 of KMP.

which are the focus of traditional return prediction models. Off-diagonals track cross-predictability phenomena.

PPA applies SVD to the prediction matrix. Kelly *et al.* (2020a) prove that the singular vectors of Π—those that account for the lion's share of covariation between signals and future returns—are a set of normalized investment portfolios, ordered from those most predictable by S to those least predictable.

The leading singular vectors of Π are "principal portfolios." Kelly *et al.* (2020a) show that principal portfolios have a direct interpretation as optimal portfolios. Specifically, they are the most "timeable" portfolios based on signal S and they offer the highest average returns for an investor that faces a leverage constraint (i.e., one who cannot hold arbitrarily large positions).

Kelly *et al.* (2020a) also point out that useful information about asset pricing models and pricing errors is encoded in Π. To tease out return predictability associated with beta versus alpha, we can decompose Π into its symmetric part (Π^s) and anti-symmetric part (Π^a):

$$\Pi = \underbrace{\frac{1}{2}(\Pi + \Pi')}_{\Pi^s} + \underbrace{\frac{1}{2}(\Pi - \Pi')}_{\Pi^a} . \tag{3.14}$$

Kelly *et al.* (2020a) prove that the leading singular vectors of Π^a ("principal alpha portfolios") have an interpretation as pure-alpha strategies while those of Π^s have an interpretation as pure-beta portfolios ("principal exposure portfolios"). Therefore, Kelly *et al.* (2020a) propose a new test of asset pricing models based on the average returns of principal alpha portfolios (as an alternative to tests such as Gibbons *et al.*, 1989). While Kelly *et al.* (2020a) focus on the case of a single signal for each asset, He *et al.* (2022b) show how to tractably extend this to multiple signals. Goulet Coulombe and Göbel, 2023 approach a similar problem as Kelly *et al.* (2020a)—constructing a portfolio that is maximally predictable—using a combination of random forest and constrained ridge regression.

Kelly *et al.* (2020a) apply PPA to a number of data sets across asset classes. Figure 3.3 presents one example from their analysis. The set of assets R_{t+1} are the 25 Fama-French size and book-to-market portfolios, and the signals S_t are time series momentum for each asset. The figure shows large out-of-sample Sharpe ratios of the resulting principal portfolios, and shows that they generate significant alpha versus a benchmark model that includes the five Fama-French factors plus a standard time series momentum factor.

3.6 Decision Trees

Modern asset pricing models (with habit persistence, long-run risks, or time-varying disaster risk) feature a high degree of state dependence in financial market behaviors, suggesting that interaction effects are potentially important to include in empirical models. For example, Hong *et al.* (2000) formulate an information theory in which the momentum effect is modulated by a firm's size and its extent of analyst coverage, emphasizing that expected stock returns vary with interactions among firm characteristics. Conceptually, it is straightforward to incorporate such effects in linear models by introducing variable interactions, just like equation (3.5) introduces nonlinear transformations. The issue is that, lacking a priori assumptions about the relevant interactions, this generalized additive approach quickly runs up against computational limits because multi-way interactions increase the number of parameters combinatorially.[7]

Regression trees provide a way to incorporate multi-way predictor interactions at much lower computational cost. Trees partition data observations into groups that share common feature interactions. The logic is that, by finding homogenous groups of observations, one can use past data for a given group to forecast the behavior of a new observation that arrives in the group. Figure 3.4 shows an example with two predictors, "size" and "b/m." The left panel describes how the tree assigns each observation to a partition based on its predictor values. First, observations are sorted on size. Those above the breakpoint of 0.5 are assigned to Category 3. Those with small size are then further sorted by b/m. Observations with small size and b/m below 0.3 are assigned to Category 1, while those with b/m above 0.3 go into Category 2. Finally, forecasts for observations in each partition are defined as the simple average of the outcome variable's value among observations in that partition.

The general prediction function associated with a tree of K "leaves" (terminal nodes) and depth L is

$$g(z_{i,t};\theta,K,L)=\sum_{k=1}^{K}\theta_k\mathbf{1}_{\{z_{i,t}\in C_k(L)\}}, \tag{3.15}$$

where $C_k(L)$ is one of the K partitions. Each partition is a product of up to L indicator functions. The constant parameter θ_k associated with partition k is estimated as the sample average of outcomes within the partition.

The popularity of decision trees stems less from their structure and more in the "greedy" algorithms that can effectively isolate highly predictive partitions at low computational cost. While the specific predictor variable upon which a branch is split (and the specific value where the split occurs) is chosen to minimize forecast error, the space of

[7] As Gu *et al.* (2020b) note, *"Parameter penalization does not solve the difficulty of estimating linear models when the number of predictors is exponentially larger than the number of observations. Instead, one must turn to heuristic optimization algorithms such as stepwise regression (sequentially adding/dropping variables until some stopping rule is satisfied), variable screening (retaining predictors whose univariate correlations with the prediction target exceed a certain value), or others."*

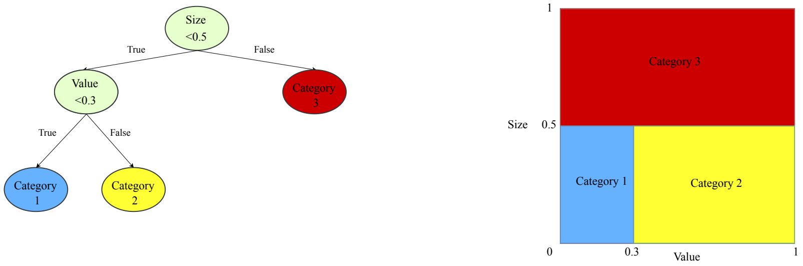


Figure 3.4: Regression Tree Example

Source: Gu *et al.* (2020b).
Note: This figure presents the diagrams of a regression tree (left) and its equivalent representation (right) in the space of two characteristics (size and value). The terminal nodes of the tree are colored in blue, yellow, and red, respectively. Based on their values of these two characteristics, the sample of individual stocks is divided into three categories.

possible splits is so expansive that the tree cannot be globally optimized. Instead, splits are determined myopically and the estimated tree is a coarse approximation of the infeasible best tree model.

Trees flexibly accommodate interactions (a tree of depth L can capture $(L-1)$-way interactions) but are prone to overfit. To counteract this, trees are typically employed in regularized “ensembles.” One common ensembling method is “boosting” (gradient boosted regression trees, “GBRT”), which recursively combines forecasts from many trees that are individually shallow and weak predictors but combine into a single strong predictor (see Schapire, 1990; Friedman, 2001). The boosting procedure begins by fitting a shallow tree (e.g., with depth $L = 1$). Then, a second simple tree fits the prediction residuals from the first tree. The forecast from the second tree is shrunk by a factor $\nu \in (0, 1)$ then added to the forecast from the first tree. Shrinkage helps prevent the model from overfitting the preceding tree’s residuals. This is iterated into an additive ensemble of B shallow trees. The boosted ensemble thus has three tuning parameters (L, ν, B).

Rossi and Timmermann (2015) investigate Merton (1973)’s ICAPM to evaluate whether the conditional equity risk premium varies with the market’s exposure to economic state variables. The lynchpin to this investigation is an accurate measurement of the ICAPM conditional covariances. To solve this problem, the authors use boosted trees to predict the realized covariance between a daily index of aggregate economic activity and the daily return on the market portfolio. The predictors include a collection of macro-finance data series from Welch and Goyal (2008) as well as past realized covariances. In contrast to prior literature, these conditional covariance estimates have a significantly positive times series association with the conditional equity risk premium, and imply economically plausible magnitudes of investor risk aversion. The authors show that linear models for the conditional covariance are badly misspecified, and this is likely responsible for mixed results in prior tests of the ICAPM. They attribute their positive conclusion regarding the ICAPM to the boosted tree methodology, whose nonlinear flexibility reduces misspecification of the conditional covariance function.

In related work, Rossi (2018) uses boosted regression trees with macro-finance predictors to directly forecast aggregate stock returns (and volatility) at the monthly frequency, but without imposing the ICAPM's restriction that predictability enters through conditional variance and covariances with economic state variables. He shows that boosted tree forecasts generate a monthly return prediction R^2 of 0.3% per month out-of-sample (compared to -0.7% using a historical mean return forecast), with directional accuracy of 57.3% per month. This results in a significant out-of-sample alpha versus a market buy-and-hold strategy and a corresponding 30% utility gain for a mean-variance investor with risk aversion equal to two.

A second popular tree regularizer is the "random forest" model which, like boosting, creates an ensemble of forecasts from many shallow trees. Following the more general "bagging" (bootstrap aggregation) procedure of Breiman (2001), random forest draws B bootstrap samples of the data, fits a separate regression tree to each, then averages their forecasts. In addition to randomizing the estimation samples via bootstrap, random forest also randomizes the set of predictors available for building the tree (an approach known as "dropout"). Both the bagging and dropout components of random forest regularize its forecasts. The depth L of the individual trees, the number of bootstrap samples B, and the dropout rate are tuning parameters.

Tree-based return predictions are isomorphic to conditional (sequential) sorts similar to those used in the asset pricing literature. Consider a sequential tercile sort on stock size and value signals to produce nine double-sorted size and value portfolios. This is a two-level ternary decision tree with first-layer split points equal to the terciles of the size distribution and second-layer split points at terciles of the value distribution. Each leaf j of the tree can be interpreted as a portfolio whose time t return is the (weighted) average of returns for all stocks allocated to leaf j in time t. Likewise, the forecast for each stock in leaf j is the average return among stocks allocated to j in the training set prior to t.

Motivated by this isomorphism, Moritz and Zimmermann (2016) conduct conditional portfolio sorts by estimating regression trees from a large collection of stock characteristics, rather than using pre-defined sorting variables and break points. While traditional sorts can accommodate at most two or three way interactions among stock signals, trees can search more broadly for the most predictive multi-way interactions. Rather than conducting a single tree-based sort, they use random forest to produce ensemble return forecasts from 200 trees. The authors report that an equal-weight long-short decile spread strategy based on one-month-ahead random forest forecasts earns a monthly return of 2.3% (though the authors also show that this performance is heavily influenced by the costly-to-trade one-month reversal signal).

Building on Moritz and Zimmermann (2016), Bryzgalova *et al.* (2020) use a method they call “AP-trees” to conduct portfolio sorts. Whereas Moritz and Zimmermann (2016) emphasize the return prediction power of tree-based sorts, Bryzgalova *et al.* (2020) emphasize the usefulness of sorted portfolios themselves as test assets for evaluating asset pricing models. AP-trees differ from traditional tree-based models in that it does not learn the tree structure per se, but instead generates it using median splits with a pre-selected ordering of signals. This is illustrated on the left side of Figure 3.5 for a three-level tree that is specified (not estimated) to split first by median value, then again by median value (i.e., produce a quartile splits), and finally by median firm size. AP-trees introduce “pruning” according to a global Sharpe ratio criterion. In particular, from the pre-determined tree structure, each node in the tree constitutes a portfolio. The AP-trees procedure finds the mean-

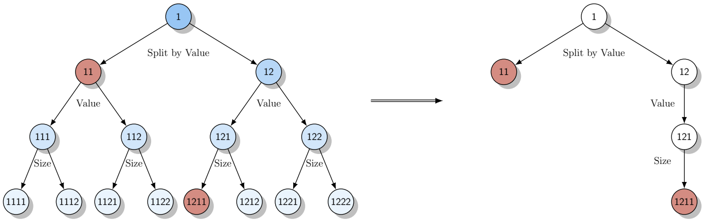


Figure 3: Illustration of pruning with multiple characteristics

The figure shows sample trees, original and pruned for portfolios of depth 3, and constructed based on size and book-to-market as the only characteristics. The fully pruned set of portfolios is based on eight trees, where the right figure illustrates a potential outcome for one tree.

Figure 3.5: Source: Bryzgalova *et al.* (2020)

variance efficient combination of node portfolios, imposing an elastic net penalty on the combination weights. Finally, the procedure discards any nodes that receive zero weight in the cross-validated optimization, while the surviving node portfolios are used as test assets in auxiliary asset pricing analyses. The authors repeat this analysis for many triples of signals, constructing a variety of pruned AP-trees to use as test assets. Bryzgalova *et al.* (2020) highlight the viability of their optimized portfolio combinations as a candidate stochastic discount factor, linking it to the literature in Section 5.5.[8]

Cong *et al.* (2022) push the frontier of asset pricing tree methods by introducing the Panel Tree (P-Tree) model, which splits the cross-section of stock returns based on firm characteristics to develop a conditional asset pricing model. At each split, the algorithm selects a split rule candidate (such as size < 0.2) from the pool of characteristics (normalized to the range -1 to 1) and a predetermined set of split thresholds (e.g., -0.6, -0.2, 0.2, 0.6) to form child leaves, each representing a portfolio of returns. They propose a compelling split criterion based on an asset pricing objective. The P-tree algorithm terminates when a pre-determined minimal leaf size or maximal number of leaves is

[8] Giglio *et al.* (2021b) point out that test asset selection is a remedy for weak factors. Therefore, creating AP-trees without pruning is also a reasonable recipe for constructing many well-diversified portfolios to serve as test assets which can be selected to construct hedging portfolios for weak factors.

met. The P-tree procedure yields a one-factor conditional asset pricing model that retains interpretability and flexibility via a single tree-based structure.

Tree methods are used in a number of financial prediction tasks beyond return prediction. We briefly discuss three examples: credit risk prediction, liquidity prediction, and volatility prediction. Correia *et al.* (2018) use a basic classification tree to forecast credit events (bankruptcy and default). They find that prediction accuracy benefits from a wide collection of firm-level risk features, and they demonstrate superior out-of-sample classification accuracy from their model compared to traditional credit risk models (e.g. Altman, 1968; Ohlson, 1980). Easley *et al.* (2020) use random forest models to study the high frequency dynamics of liquidity and risk in a truly big data setting—tick data for 87 futures markets. They show that a variety of market liquidity measures (such as Kyle's lambda, the Amihud measure, and the probability of informed trading) have substantial predictive power for subsequent directional moves in risk and liquidity outcomes (including bid-ask spread, return volatility, and return skewness). Mittnik *et al.* (2015) use boosted trees to forecast monthly stock market volatility. They use 84 macro-finance time series as predictors and use boosting to recursively construct an ensemble of volatility prediction trees that optimizes the Gaussian log likelihood for monthly stock returns. The authors find large and statistically significant reductions in prediction errors from tree-based volatility forecasts relative to GARCH and EGARCH benchmark models.

3.7 Vanilla Neural Networks

Neural networks are perhaps the most popular and most successful models in machine learning. They have theoretical underpinnings as "universal approximators" for any smooth predictive function (Hornik *et al.*, 1989; Cybenko, 1989). They suffer, however, from a lack of transparency and interpretability.

Gu *et al.* (2020b) analyze predictions from "feed-forward" networks. We discuss this structure in detail as it lays the groundwork before more sophisticated architectures such as recurrent and convolutional networks

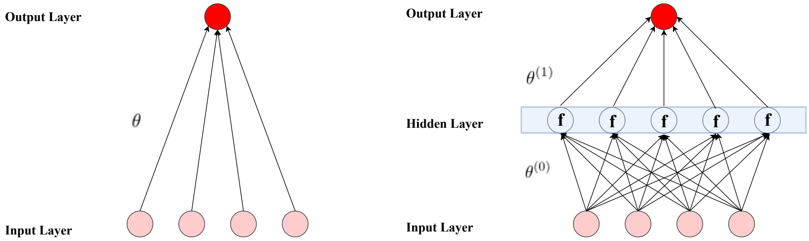


Figure 3.6: Neural Networks

Note: This figure provides diagrams of two simple neural networks with (right) or without (left) a hidden layer. Pink circles denote the input layer and dark red circles denote the output layer. Each arrow is associated with a weight parameter. In the network with a hidden layer, a nonlinear activation function f transforms the inputs before passing them on to the output.

(discussed later in this review). These consist of an "input layer" of raw predictors, one or more "hidden layers" that interact and nonlinearly transform the predictors, and an "output layer" that aggregates hidden layers into an ultimate outcome prediction.

Figure 3.6 shows two example networks. The left panel shows a very simple network that has no hidden layers. The predictors (denoted $z_1, ..., z_4$) are weighted by a parameter vector (θ) that includes an intercept and one weight parameter per predictor. These weights aggregate the signals into the forecast $\theta_0 + \sum_{k=1}^{4} z_k\theta_k$. As the example makes clear, without intermediate nodes a neural network is a linear regression model.

The right panel introduces a hidden layer of five neurons. A neuron receives a linear combination of the predictors and feeds it through a nonlinear "activation function" f, then this output is passed to the next layer. E.g., output from the second neuron is $x_2^{(1)} = f\left(\theta_{2,0}^{(0)} + \sum_{j=1}^{4} z_j\theta_{2,j}^{(0)}\right)$. In this example, the results from each neuron are linearly aggregated into a final output forecast:

$$g(z;\theta) = \theta_0^{(1)} + \sum_{j=1}^{5} x_j^{(1)}\theta_j^{(1)}.$$

There are many choices for the nonlinear activation function (such as sigmoid, sine, hyperbolic, softmax, ReLU, and so forth). "Deep" feed-forward networks introduce additional hidden layers in which nonlinear transformations from a prior hidden layer are combined in a linear combination and transformed once again via nonlinear activation. This iterative nonlinear composure produces richer and more highly parameterized approximating models.

As in Gu *et al.* (2020b), a deep feed-forward neural network has the following general formula. Let $K^{(l)}$ denote the number of neurons in each layer $l = 1, ..., L$. Define the output of neuron k in layer l as $x_k^{(l)}$. Next, define the vector of outputs for this layer (augmented to include a constant, $x_0^{(l)}$) as $x^{(l)} = (1, x_1^{(l)}, ..., x_{K^{(l)}}^{(l)})'$. To initialize the network, similarly define the input layer using the raw predictors, $x^{(0)} = (1, z_1, ..., z_N)'$. The recursive output formula for the neural network at each neuron in layer $l > 0$ is then

$$x_k^{(l)} = f\left(x^{(l-1)\prime}\theta_k^{(l-1)}\right), \tag{3.16}$$

with final output

$$g(z; \theta) = x^{(L-1)\prime}\theta^{(L-1)}. \tag{3.17}$$

The number of weight parameters in each hidden layer l is $K^{(l)}(1 + K^{(l-1)})$, plus another $1 + K^{(L-1)}$ weights for the output layer. The five-layer network of Gu *et al.* (2020b), for example, has 30,185 parameters.

Gu *et al.* (2020b) estimate monthly stock-level panel prediction models for the CRSP sample from 1957 to 2016. Their raw features include 94 rank-standardized stock characteristic interacted with eight macro-finance time series as well as 74 industry indicators for a total of 920 features. They infer the trade-offs of network depth in the return forecasting problem by analyzing the performance of networks with one to five hidden layers (denoted NN1 through NN5). The monthly out-of-sample prediction R^2 is 0.33%, 0.40%, and 0.36% for NN1, NN3, and NN5 models, respectively. This compares with an R^2 of 0.16% from a benchmark three signal (size, value, and momentum) linear model advocated by Lewellen (2015). For the NN3 model, the R^2 is notably higher for large caps at 0.70%, indicating that machine learning is not merely picking up small scale inefficiencies driven by illiquidity. The out-of-sample R^2 rises to 3.40% when forecasting annual returns rather than monthly, illustrating that the neural networks are also able to isolate predictable patterns that persist over business cycle frequencies.

Table 3.3: Performance of Machine Learning Portfolios

Panel A: Equal Weights

	NN1				NN3				NN5			
	Pred	Avg	Std	SR	Pred	Avg	Std	SR	Pred	Avg	Std	SR
L	-0.45	-0.78	7.43	-0.36	-0.31	-0.92	7.94	-0.40	-0.08	-0.83	7.92	-0.36
2	0.15	0.22	6.24	0.12	0.22	0.16	6.46	0.09	0.33	0.24	6.64	0.12
3	0.43	0.47	5.55	0.29	0.45	0.44	5.40	0.28	0.51	0.53	5.65	0.32
4	0.64	0.64	5.00	0.45	0.60	0.66	4.83	0.48	0.62	0.59	4.91	0.41
5	0.80	0.80	4.76	0.58	0.73	0.77	4.58	0.58	0.71	0.68	4.56	0.51
6	0.95	0.85	4.63	0.63	0.85	0.81	4.47	0.63	0.80	0.76	4.43	0.60
7	1.12	0.84	4.66	0.62	0.97	0.86	4.62	0.64	0.88	0.88	4.60	0.66
8	1.32	0.88	4.95	0.62	1.12	0.93	4.82	0.67	1.01	0.95	4.90	0.67
9	1.63	1.17	5.62	0.72	1.38	1.18	5.51	0.74	1.25	1.17	5.60	0.73
H	2.43	2.13	7.34	1.00	2.28	2.35	8.11	1.00	2.08	2.27	7.95	0.99
H-L	2.89	2.91	4.72	2.13	2.58	3.27	4.80	2.36	2.16	3.09	4.98	2.15

Panel B: Value Weights

	NN1				NN3				NN5			
	Pred	Avg	Std	SR	Pred	Avg	Std	SR	Pred	Avg	Std	SR
L	-0.38	-0.29	7.02	-0.14	-0.03	-0.43	7.73	-0.19	-0.23	-0.51	7.69	-0.23
2	0.16	0.41	5.89	0.24	0.34	0.30	6.38	0.16	0.23	0.31	6.10	0.17
3	0.44	0.51	5.07	0.35	0.51	0.57	5.27	0.37	0.45	0.54	5.02	0.37
4	0.64	0.70	4.56	0.53	0.63	0.66	4.69	0.49	0.60	0.67	4.47	0.52
5	0.80	0.77	4.37	0.61	0.71	0.69	4.41	0.55	0.73	0.77	4.32	0.62
6	0.95	0.78	4.39	0.62	0.79	0.76	4.46	0.59	0.85	0.86	4.35	0.68
7	1.11	0.81	4.40	0.64	0.88	0.99	4.77	0.72	0.96	0.88	4.76	0.64
8	1.31	0.75	4.86	0.54	1.00	1.09	5.47	0.69	1.11	0.94	5.17	0.63
9	1.58	0.96	5.22	0.64	1.21	1.25	5.94	0.73	1.34	1.02	6.02	0.58
H	2.19	1.52	6.79	0.77	1.83	1.69	7.29	0.80	1.99	1.46	7.40	0.68
H-L	2.57	1.81	5.34	1.17	1.86	2.12	6.13	1.20	2.22	1.97	5.93	1.15

Note: In this table, we report the performance of prediction-sorted portfolios over the 1987-2016 testing period (trained on data from 1957-1974 and validated on data from 1975-1986). All stocks are sorted into deciles based on their predicted returns for the next month. Column "Pred", "Avg", "Std", and "SR" provide the predicted monthly returns for each decile, the average realized monthly returns, their standard deviations, and Sharpe ratios, respectively.

Next, Gu *et al.* (2020b) report decile portfolios sorts based on neural network monthly return predictions, recreated in Table 3.3. Panel A reports equal-weight average returns and Panel B reports value-weight returns. Out-of-sample portfolio returns increases monotonically across deciles. The quantitative match between predicted returns and average realized returns using neural networks is impressive. A long-short decile spread portfolio earns an annualized equal-weight Sharpe ratio of 2.1, 2.4, and 2.2 for NN1, NN3, and NN5, respectively. All three earn value-weight Sharpe ratios of 1.2. So, while the return prediction R^2 is higher for large stocks, the return prediction content of the neural network models is especially profitable when used to trade small stocks. Avramov *et al.* (2022a) corroborate this finding with a more thorough investigation of the limits-to-arbitrage that impinge on machine learning trading strategies. They demonstrate that neural network predictions are most successful among difficult-to-value and difficult-to-arbitrage stocks. They find that, after adjusting for trading costs and other practical considerations, neural network-based strategies remain significantly beneficial relative to typical benchmarks. They are highly profitable (particularly among long positions), have less downside risk, and continue to perform well in recent data.

As a frame of reference, Gu *et al.* (2020b)'s replication of Lewellen (2015)'s three-signal linear model earns an equal-weight Sharpe ratio of 0.8 (0.6 value-weight). This is impressive in its own right, but the large improvement from neural network predictions emphasizes the important role of nonlinearities and interactions in expected return models. Figure 3.7 from Gu *et al.* (2020b) illustrates this fact by plotting the effect of a few characteristics in their model. It shows how expected returns vary as pairs of characteristics are varied over their support [-1,1] while holding all other variables fixed at their median value. The effects reported are interactions of stock size (mvel1) with short-term reversal (mom1m), momentum (mom12m), total volatility (retvol) and accrual (acc). For example, the upper-left figure shows that the short-term reversal effect is strongest and is essentially linear among small stocks (blue line). But among mega-cap stocks (green line), the reversal effect is concave, manifesting most prominently when past mega-cap returns are strongly positive.

The models in Gu *et al.* (2020b) apply to the stock-level panel. Feng *et al.* (2018) use a pure time series feed-forward network to forecast aggregate market returns using the Welch-Goyal macro-finance predictors, and find significant out-of-sample R^2 gains compared to linear models

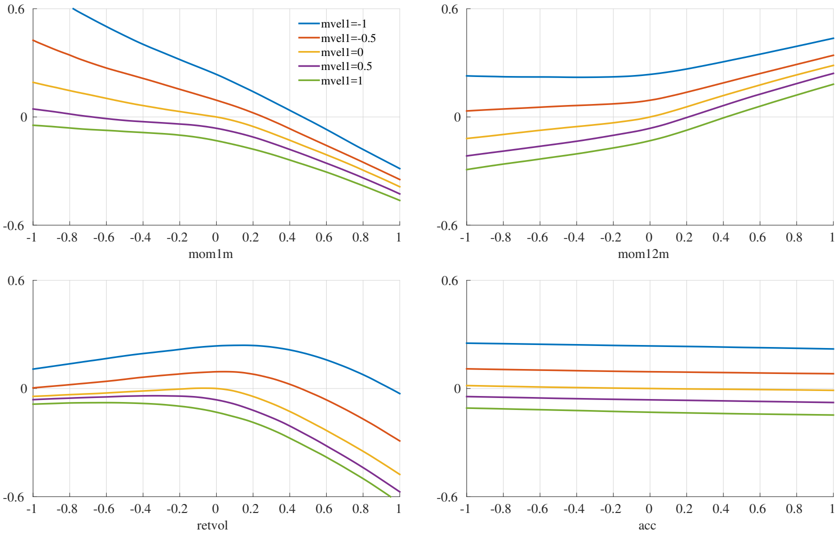


Figure 3.7: Expected Returns and Characteristic Interactions (NN3)

Note: Sensitivity of expected monthly percentage returns (vertical axis) to interactions effects for mvel1 with mom1m, mom12m, retvol, and acc in model NN3 (holding all other covariates fixed at their median values).

(including those with penalization and dimension reduction).

3.8 Comparative Analyses

A number of recent papers conduct comparisons of machine learning return prediction models in various data sets. In the first such study, Gu *et al.* (2020b) perform a comparative analysis of the major machine learning methods outlined above in the stock-month panel prediction setting. Table 3.4 recreates their main result for out-of-sample panel return prediction R^2 across models. This comparative analysis helps establish a number of new empirical facts for financial machine learning. First, the simple linear model with many predictors ("OLS" in the table) suffers in terms of forecast accuracy, failing to outperform a naive forecast of zero. Gu *et al.* (2020b) define their predictive R^2 relative to a naive forecast of zero rather than the more standard benchmark of

	OLS	OLS-3	PLS	PCR	ENet	GLM	RF	GBRT	NN1	NN2	NN3	NN4	NN5
	+H	+H			+H	+H		+H					
All	-3.46	0.16	0.27	0.26	0.11	0.19	0.33	0.34	0.33	0.39	0.40	0.39	0.36
Top 1000	-11.28	0.31	-0.14	0.06	0.25	0.14	0.63	0.52	0.49	0.62	0.70	0.67	0.64
Bottom 1000	-1.30	0.17	0.42	0.34	0.20	0.30	0.35	0.32	0.38	0.46	0.45	0.47	0.42

Table 3.4: Monthly Out-of-sample Stock-level Prediction Performance (Percentage R^2_{oos})

Note: In this table, Gu *et al.* (2020b) report monthly R^2_{oos} for the entire panel of stocks using OLS with all variables (OLS), OLS using only size, book-to-market, and momentum (OLS-3), PLS, PCR, elastic net (ENet), generalize linear model (GLM), random forest (RF), gradient boosted regression trees (GBRT), and neural networks with one to five layers (NN1–NN5). "+H" indicates the use of Huber loss instead of the l_2 loss. R^2_{oos}'s are also reported within subsamples that include only the top 1,000 stocks or bottom 1,000 stocks by market value.

the historical sample mean return, noting:

> *Predicting future excess stock returns with historical averages typically* underperforms *a naive forecast of zero by a large margin. That is, the historical mean stock return is so noisy that it artificially lowers the bar for "good" forecasting performance. We avoid this pitfall by benchmarking our R^2 against a forecast value of zero. To give an indication of the importance of this choice, when we benchmark model predictions against historical mean stock returns, the out-of-sample monthly R^2 of all methods rises by roughly three percentage points.*[9]

Regularizing with either dimension reduction or shrinkage improves the R^2 of the linear model to around 0.3% per month. Nonlinear models, particularly neural networks, help even further, especially among large cap stocks. Nonlinearities outperform a low-dimensional linear model that uses only three highly selected predictive signals from the literature ("OLS-3" in the table).

Nonlinear models also provide large incremental benefits in economic terms. Table 3.5 recreates Table 8 of Gu *et al.* (2020b), showing

[9] He *et al.* (2022a) propose an alternative cross-section out-of-sample R^2 that uses the cross section mean return as the naïve forecast.

	OLS-3	PLS	PCR	ENet	GLM	RF	GBRT	NN1	NN2	NN3	NN4	NN5
	+H			+H	+H		+H					
Risk-adjusted Performance (Value Weighted)												
Mean Ret.	0.94	1.02	1.22	0.60	1.06	1.62	0.99	1.81	1.92	1.97	2.26	2.12
FF5+Mom α	0.39	0.24	0.62	-0.23	0.38	1.20	0.66	1.20	1.33	1.52	1.76	1.43
t-stats	2.76	1.09	2.89	-0.89	1.68	3.95	3.11	4.68	4.74	4.92	6.00	4.71
R^2	78.60	34.95	39.11	28.04	30.78	13.43	20.68	27.67	25.81	20.84	20.47	18.23
IR	0.54	0.21	0.57	-0.17	0.33	0.77	0.61	0.92	0.93	0.96	1.18	0.92
Risk-adjusted Performance (Equally Weighted)												
Mean Ret.	1.34	2.08	2.45	2.11	2.31	2.38	2.14	2.91	3.31	3.27	3.33	3.09
FF5+Mom α	0.83	1.40	1.95	1.32	1.79	1.88	1.87	2.60	3.07	3.02	3.08	2.78
$t(\alpha)$	6.64	5.90	9.92	4.77	8.09	6.66	8.19	10.51	11.66	11.70	12.28	10.68
R^2	84.26	26.27	40.50	20.89	21.25	19.91	11.19	13.98	10.60	9.63	11.57	14.54
IR	1.30	1.15	1.94	0.93	1.58	1.30	1.60	2.06	2.28	2.29	2.40	2.09

Table 3.5: Risk-adjusted Performance, Drawdowns, and Turnover of Machine Learning Portfolios

Note: The tables reports average monthly returns in percent as well as alphas, information ratios (IR), and R^2 with respect to the Fama-French five-factor model augmented to include the momentum factor.

out-of-sample performance of long-short decile spread portfolios sorted on return forecasts from each model. First, it is interesting to note that models with relative small out-of-sample R^2 generates significant trading gains, in terms of alpha and information ratio relative to the Fama-French six-factor model (including a momentum factor). This is consistent with the Kelly *et al.* (2022a) point that R^2 is an unreliable diagnostic of the economic value of a return prediction; they instead recommend judging financial machine learning methods based on economic criteria (such as trading strategy Sharpe ratio). Nonlinear models in Table 3.5 deliver the best economic value in terms of trading strategy performance.

Subsequent papers conduct similar comparative analyses in other asset classes. Choi *et al.* (2022) analyze the same models as Gu *et al.* (2020b) in international stock markets. They reach similar conclusions that the best performing models are nonlinear. Interestingly, they demonstrate the viability of transfer learning. In particular, a model trained on US data delivers significant out-of-sample performance when used to forecast international stock returns. Relatedly, Jiang *et al.* (2018) find largely similar patterns between stock returns and firm characteristics

	OLS	PCA	PLS	Lasso	Ridge	ENet	RF	FFN	LSTM	Combination
R^2_{oos}	−3.36	2.07	2.03	1.85	1.89	1.87	2.19	2.37	2.28	2.09
Avg. Returns	0.16	0.51	0.63	0.39	0.33	0.43	0.79	0.75	0.79	0.67

Table 3.6: Machine Learning Comparative Bond Return Prediction (Bali *et al.*, 2020)

Note: The first row reports out-of-sample R^2 in percentage for the entire panel of corporate bonds using the 43 bond characteristics (from Table 2 of Bali *et al.* (2020)). The second row reports the average out-of-sample monthly percentage returns of value-weighted decile spread bond portfolios sorted on machine learning return forecasts (from Table 3 of Bali *et al.* (2020)).

in China using PCR and PLS. Recently, Leippold *et al.* (2022) compare machine learning models for predicting Chinese equity returns and highlight how liquidity, retail investor participation, and state-owned enterprises play a pronounced role in Chinese market behavior.

Ait-Sahalia *et al.* (2022) study the predictability of high-frequency returns (and other quantities like duration and volume) using machine learning methods. They construct 13 predictors over 9 different time windows, resulting in a total of 117 variables. They experiment with S&P 100 index components over a sample of two years and find that all methods have very similar performance, except for OLS. The improvements from a nonlinear method like random forest or boosted trees are limited for most cases when compared with lasso.

Bali *et al.* (2020) and He *et al.* (2021) conduct comparative analyses of machine learning methods for US corporate bond return prediction. The predictive signals used by Bali *et al.* (2020) include a large set of 43 bond characteristics such as issuance size, credit rating, duration, and so forth. The models they study are the same as Gu *et al.* (2020b) plus an LSTM network (we discuss LSTM in the next subsection). Table 3.6 reports results of Bali *et al.* (2020). Their comparison of machine learning models in terms of predictive R^2 and decile spread portfolio returns largely corroborate the conclusions of Gu *et al.* (2020b). The unregularized linear model is the worst performer. Penalization and dimension reduction substantially improve linear model performance. And nonlinear models are the best performers overall.

In addition, Bali *et al.* (2020) investigate a machine learning prediction framework that respects no-arbitrage implications for consistency between a firm's equity and debt prices. This allows them to form predictions across the capital structure of a firm—leveraging equity information to predict bond returns. They find that

> *"once we impose the Merton (1974) model structure, equity characteristics provide significant improvement above and beyond bond characteristics for future bond returns, whereas the incremental power of equity characteristics for predicting bond returns are quite limited in the reduced-form approach when such economic structure is not imposed."*

This is a good example of the complementarity between machine learning and economic structure, echoing the argument of Israel *et al.* (2020).

Lastly, Bianchi *et al.* (2021) conduct a comparative analysis of machine learning models for predicting US government bond returns. This is a pure time series environment (cf. comparative analyses discussed above which study panel return data). Nonetheless, Bianchi *et al.* (2021) reach similar conclusions regarding the relative merits of penalized and dimension-reduced linear models over unconstrained linear models, and of nonlinear models over linear models.

3.9 More Sophisticated Neural Networks

Recurrent neural networks (RNNs) are popular models for capturing complex dynamics in sequence data. They are, in essence, highly parameterized nonlinear state space models, making them naturally interesting candidates for time series prediction problems. A promising use of RNNs is to extend the restrictive model specification given by (3.2) to capture longer range dependence between returns and characteristics. Specifically, we consider a general recurrent model for the expected return of stock i at time t:

$$\mathrm{E}_t[R_{i,t+1}] = g^\star(h_{i,t}), \tag{3.18}$$

where $h_{i,t}$ is a vector of hidden state variables that depend on $z_{i,t}$ and its past history.

The canonical RNN assumes that

$$g^\star(h_{i,t}) = \sigma(c + Vh_{i,t}), \quad h_{i,t} = \tanh(b + Wh_{i,t-1} + Uz_{i,t}),$$

where b, c, U, V, and W are unknown parameters, and $\sigma(\cdot)$ is a sigmoid function. The above equation includes just one layer of hidden states. It is straightforward to stack multiple hidden state layers together to construct a deep RNN that accommodates more complex sequences. For instance, we can write $h_{i,t}^{(0)} = z_{i,t}$, and for $1 \leq l \leq L$, we have

$$g^{\star}(h_{i,t}^{(L)}) = \sigma(c + Vh_{i,t}^{(L)}), \quad h_{i,t}^{(l)} = \tanh(b + Wh_{i,t-1}^{(l)} + Uh_{i,t}^{(l-1)}).$$

The canonical RNN struggles to capture long-range dependence. Its telescoping structure implies exponentially decaying weights of lagged states on current states so the long-range gradient flow vanishes rapidly in the learning process.

Hochreiter and Schmidhuber (1997) propose a special form of RNN known as the long-short-term-memory (LSTM) model. It accommodates a mixture of short-range and long-range dependence through a series of gate functions that control information flow from h_{t-1} to h_t (we abbreviate the dependence on i for notation simplicity):

$$h_t = o_t \odot \tanh(c_t), \quad c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \tag{3.19}$$

where $\odot$ denotes element-wise multiplication, c_t is so-called cell state, i_t, o_t, and f_t are "gates" that are themselves sigmoid functions of z_t and h_{t-1}:

$$a_t = \sigma(W_a z_t + U_a h_{t-1} + b_a), \tag{3.20}$$

where a can be either gate i, o, or f, and W_a, U_a, and b_a are parameters.

The cell state works like a conveyor belt, representing the "memory" unit of the model. Past values of c_{t-1} contributes additively to c_t (barring from some adjustment by f_t). This mechanism enables the network to memorize long-range information. Meanwhile, f_t controls how much information from the past to forget, hence it is called the forget gate. Fresh information from h_{t-1} and z_t arrive through $\tilde{c}_t$:

$$\tilde{c}_t = \tanh(W_c z_t + U_c h_{t-1} + b_c), \tag{3.21}$$

which is injected into the memory cell c_t. The input gate, i_t, controls the content of $\tilde{c}_t$ to be memorized. Finally, o_t determines which information is passed on to the next hidden state h_t, and is hence called output gate.

Intuitively, not all information memorized by c_t needs be extracted for prediction (based on h_t).

Despite their usefulness for time series modeling, LSTM and related methods (like the gated recurrent unit of Cho *et al.*, 2014) have seen little application in the empirical finance literature. We have discussed a notable exception by Bali *et al.* (2020) above for predicting corporate bond returns. (Guijarro-Ordonez *et al.*, 2022) use RNN architectures to predict daily stock returns. Another example is Cong *et al.* (2020), who compare simple feed-forward networks to LSTM and other recurrent neural networks in monthly stock return prediction. They find that predictions from their LSTM specification outperform feed-forward counterparts, though the description of their specifications is limited and their analysis is conducted more in line with the norms of the computer science literature. Indeed, there is a fairly voluminous computer science literature using a wide variety of neural networks to predict stock returns (e.g. Sezer *et al.*, 2020). The standard empirical analysis in this literature aims to demonstrate basic proof of concept through small scale illustrative experiments and tend to focus on high frequencies (daily or intra-daily, rather than the perhaps more economically interesting frequencies of months or years) or tend to analyze one or a few assets at a time (as opposed to a more representative cross section of assets).[10] This is in contrast to the more extensive empirical analyses common in the finance and economics literature that tend to analyze monthly or annual data for large collections of assets.

3.10 Return Prediction Models For “Alternative” Data

Alternative (or more colloquially, “alt”) data has become a popular topic in the asset management industry, and recent research has made strides developing machine learning models for some types of alt data. We discuss two examples in this section, text data and image data, and some supervised machine learning models customized to these alt data sources.

[10] Examples include Rather *et al.* (2015), Singh and Srivastava (2017), Chong *et al.* (2017), Bao *et al.* (2017).

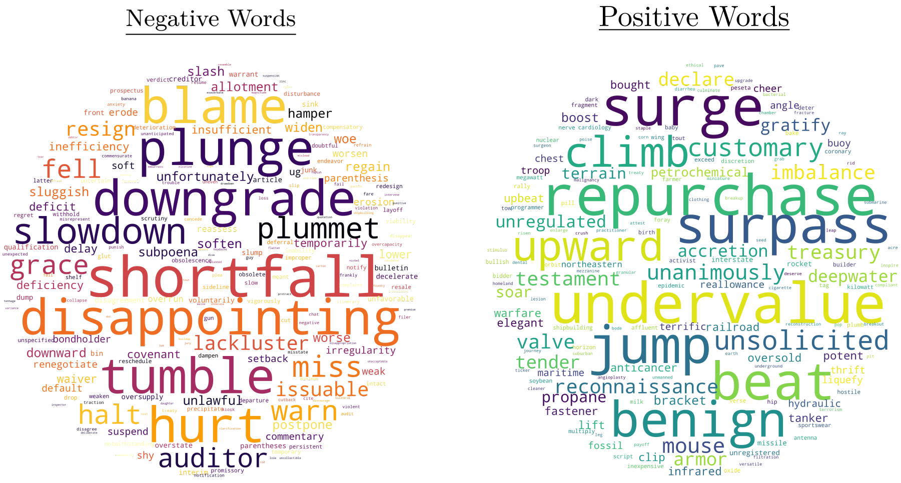
3.10.1 Textual Analysis

Textual analysis is among the most exciting and fastest growing frontiers in finance and economics research. The early literature evolved from “close” manual reading by researchers (e.g. Cowles, 1933) to dictionary-based sentiment scoring methods (e.g. Tetlock, 2007). This literature is surveyed by Das *et al.* (2014), Gentzkow *et al.* (2019), and Loughran and McDonald (2020). In this section, we focus on text-based supervised learning with application to financial prediction.

Jegadeesh and Wu (2013), Ke *et al.* (2019), and Garcia *et al.* (2022) are examples of supervised learning models customized to the problem of return prediction using a term count or “bag of words” (BoW) representation of text documents. Ke *et al.* (2019) describe a joint probability model for the generation of a news article about a stock and that stock’s subsequent return. An article is indexed by a single underlying “sentiment” parameter that determines the article’s tilt toward good or bad news about the stock. This same parameter predicts the direction of the stock’s future return. From a training sample of news articles and associated returns, Ke *et al.* (2019) estimate the set of most highly sentiment-charged (i.e., most return-predictive) terms and their associated sentiment values (i.e., their predictive coefficients). In essence, they provide a data-driven methodology for constructing sentiment dictionaries that are customized to specific supervised learning tasks.

Their method, called “SESTM” (Sentiment Extraction via Screening and Topic Modeling), has three central components. The first step isolates the most relevant terms from a very large vocabulary of terms via predictive correlation screening. The second step assigns term-specific sentiment weights using a supervised topic model. The third step uses the estimated topic model to assign article-level sentiment scores via penalized maximum likelihood.

SESTM is easy and cheap to compute. The model itself is analytically tractable and the estimator boils down to two core equations. Their modeling approach emphasizes simplicity and interpretability. Thus, it is “white box” and easy to inspect and interpret, in contrast to many state-of-the-art NLP models built around powerful yet opaque neural


Negative Words

Positive Words

Figure 3.8: Sentiment-charged Words

Note: This figure reports the list of words in the sentiment-charged set S. Font size of a word is proportional to its average sentiment tone over all 17 training samples.

network embedding specifications. As an illustration, Figure 3.8 reports coefficient estimates on the tokens that are the strongest predictors of returns in the form of a word cloud. The cloud is split into tokens with positive or negative coefficients, and the size of each token is proportional to the magnitude of its estimated predictive coefficient.

Ke *et al.* (2019) devise a series of trading strategies to demonstrate the potent return predictive power of SESTM. In head-to-head comparisons, SESTM significantly outperforms RavenPack (a leading commercial sentiment scoring vendor used by large asset managers) and dictionary-based methods such as Loughran and McDonald (2011).

A number of papers apply supervised learning models to BoW text data to predict other financial outcomes. Manela and Moreira (2017) use support vector regression to predict market volatility. Davis *et al.* (2020) use 10-K risk factor disclosure to understand firms' differential return responses to the COVID-19 pandemic, leveraging Taddy (2013)'s multinomial inverse regression methodology. Kelly *et al.* (2018) introduce a method called hurdle distributed multinomial regression (HDMR) to improve count model specifications and use it to build a text-based index measuring health of the financial intermediary sector.

These analyses proceed in two general steps. Step 1 decides on the numerical representation of the text data. Step 2 uses the representations as data in an econometric model to describe some economic phenomenon (e.g., asset returns, volatility, and macroeconomic fundamentals in the references above).

The financial text representations referenced above have some limitations. First, all of these examples begin from a BoW representation, which is overly simplistic and only accesses the information in text that is conveyable by term usage frequency. It sacrifices nearly all information that is conveyed through word ordering or contextual relationships between terms. Second, the ultra-high dimensionality of BoW representations leads to statistical inefficiencies—Step 2 econometric models must include many parameters to process all these terms despite many of the terms conveying negligible information. Dimension reductions like LDA and correlation screening are beneficial because they mitigate the inefficiency of BoW. However, they are derived from BoW and thus do not avoid the information loss from relying on term counts in the first place. Third, and more subtly, the dimension-reduced representations are *corpus specific*. For example, when Bybee *et al.* (2020) build their topic model, the topics are estimated only from *The Wall Street Journal*, despite the fact that many topics are general language structures and may be better inferred by using additional text outside of their sample.

Jiang *et al.* (2023) move the literature a step further by constructing refined news text representations derived from so-called “large language models” (LLMs). They then use these representations to improve models of expected stock returns. LLMs are trained on large text data sets that span many sources and themes. This training is conducted by specialized research teams that perform the Herculean feat of estimating a general purpose language model with astronomical parameterization on truly big text data. LLM’s have billions of parameters (or more) and are trained on billions of text examples (including huge corpora of complete books and massive portions of the internet). But for each LLM, this estimation feat is performed once, then the estimated model is made available for distribution to be deployed by non-specialized researchers in downstream tasks.

In other words, the LLM delegates Step 1 of the procedure above to the handful of experts in the world that can best execute it. A Step 2 econometric model can then be built around LLM output. Like LDA (or even BoW), the output of a foundation model is a numerical vector representation (or "embedding") of a document. A non-specialized researcher obtains this output by feeding the document of interest through software (which is open-source in many cases). The main benefit of an LLMS in Step 1 is that it provides more sophisticated and well-trained text representations than used in the literature referenced above. This benefit comes from the expressivity of heavy nonlinear model parameterizations and from training on extensive language examples across many domains, throughout human history, and in a wide variety of languages. The transferability of LLMs make this unprecedented scale of knowledge available for finance research.

Jiang *et al.* (2023) analyze return predictions based on a news text processed through a number of LLMs including Bidirectional Encoder Representations from Transformer (BERT) of Devlin *et al.* (2018), Generative Pre-trained Transformers (GPT) of Radford *et al.* (2019), and Open Pre-trained Transformers (OPT) by Zhang *et al.* (2022). They find that predictions from pre-trained LLM embeddings outperform prevailing text-based machine learning return predictions in terms of out-of-sample trading strategy performance, and that the superior performance of LLMs stems from the fact that they can more successfully capture contextual meaning in documents.

3.10.2 Image Analysis

Much of modern machine learning has evolved around the task of image analysis and computer vision, with large gains in image-related tasks deriving from the development of convolutional neural network (CNN) models. Jiang *et al.* (2022) introduce CNN image analysis techniques to the return prediction problem.

A large finance literature investigates how past price patterns forecast future returns. The philosophical perspective underpinning these analyses is most commonly that of a hypothesis test. The researcher formulates a model of return prediction based on price trends—such as a regression of one-month-ahead returns on the average return over the previous twelve months—as a test of the weak-form efficient markets null hypothesis. Yet it is difficult to see in the literature a specific alternative hypothesis. Said differently, the price-based return predictors studied in the literature are by and large ad hoc and discovered through human-intensive statistical learning that has taken place behind the curtain of the academic research process. Jiang *et al.* (2022) reconsider the idea of price-based return predictability from a different philosophical perspective founded on machine learning. Given recent strides in understanding how human behavior influences price patterns (e.g. Barberis and Thaler, 2003; Barberis, 2018), it is reasonable to expect that prices contain subtle and complex patterns about which it may be difficult to develop specific testable hypotheses. Jiang *et al.* (2022) devise a systematic machine learning approach to elicit return predictive patterns that underly price data, rather than testing specific ad hoc hypotheses.

The challenge for such an exploration is balancing flexible models that can detect potentially subtle patterns against the desire to maintain tractability and interpretability of those models. To navigate this balance, Jiang *et al.* (2022) represent historical prices as an image and use well-developed CNN machinery for image analysis to search for predictive patterns. Their images include daily opening, high, low, and closing prices (ofter referred to as an “OHLC” chart) overlaid with a multi-day moving average of closing prices and a bar chart for daily trading volume (see Figure 3.9 for a related example from Yahoo Finance).

A CNN is designed to automatically extract features from images that are predictive for the supervising labels (which are future realized returns in the case of Jiang *et al.*, 2022). The raw data consists of pixel value arrays. A CNN typically has a few core building blocks that convert the pixel data into predictive features. The building blocks are stacked together in various telescoping configurations depending on the application at hand. They spatially smooth image contents to reduce noise and accentuate shape contours to maximize correlation of images with their labels. Parameters of the building blocks are learned as part

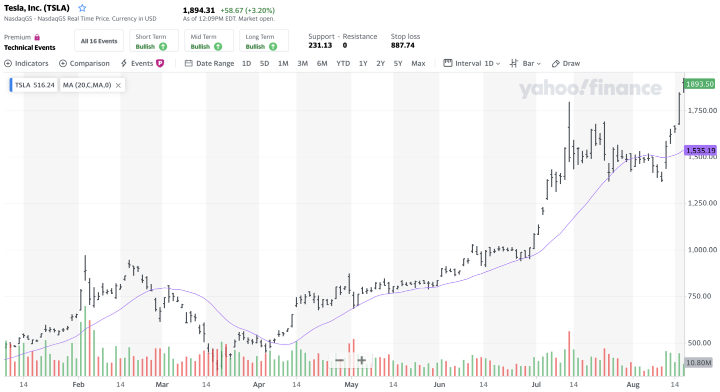


Figure 3.9: Tesla OHLC Chart from Yahoo! Finance

Note: OHLC chart for Tesla stock with 20-day moving average price line and daily volume bars. Daily data from January 1, 2020 to August 18, 2020.

of the model estimation process.

Each building block consists of three operations: convolution, activation, and pooling. “Convolution” is a spatial analogue of kernel smoothing for time series. Convolution scans through the image and, for each element in the image matrix, produces a summary of image contents in the immediately surrounding area. The convolution operates through a set of learned “filters,” which are low dimension kernel weighting matrices that average nearby matrix elements.

The second operation in a building block, “activation,” is a nonlinear transformation applied element-wise to the output of a convolution filter. For example, a “Leaky ReLU” activation uses a convex piecewise linear function, which can be thought of as sharpening the resolution of certain convolution filter output.

The final operation in a building block is “max-pooling.” This operation uses a small filter that scans over the input matrix and returns the maximum value the elements entering the filter at each location in the image. Max-pooling acts as both a dimension reduction device and as a de-noising tool.

Figure 3.10 illustrates how convolution, activation, and max-pooling combine to form a basic building block for a CNN model. By stacking many of these blocks together, the network first creates representations

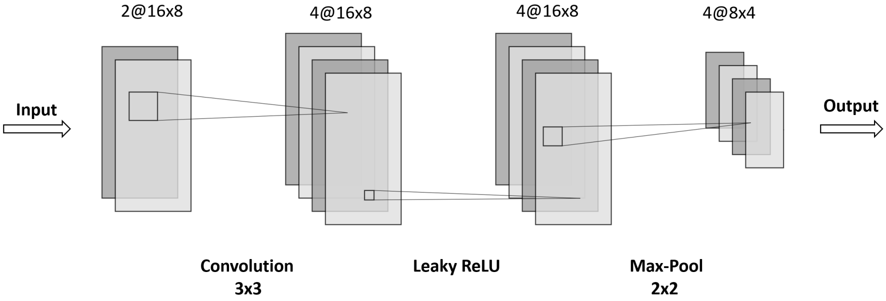


Figure 3.10: Diagram of a Building Block

Note: A building block of the CNN model consists of a convolutional layer with 3×3 filter, a leaky ReLU layer, and a 2×2 max-pooling layer. In this toy example, the input has size 16×8 with 2 channels. To double the depth of the input, 4 filters are applied, which generates the output with 4 channels. The max-pooling layer shrinks the first two dimensions (height and width) of the input by half and keeps the same depth. Leaky ReLU keeps the same size of the previous input. In general, with input of size $h \times w \times d$, the output has size $h/2 \times w/2 \times 2d$. One exception is the first building block of each CNN model that takes the grey-scale image as input: the input has depth 1 and the number of CNN filters is 32, boosting the depth of the output to 32.

of small components of the image then gradually assembles them into representations of larger areas. The output from the last building block is flattened into a vector and each element is treated as a feature in a standard, fully connected feed-forward layer for the final prediction step.[11]

Jiang *et al.* (2022) train a panel CNN model to predict the direction of future stock returns using daily US stock data from 1993 to 2019. A weekly rebalanced long-short decile spread trading strategy based on the CNN-based probability of a positive price change earns an out-of-sample annualized Sharpe ratio of 7.2 on an equal-weighted basis and 1.7 on a value-weighted basis. This outperforms well known price trend strategies—various forms of momentum and reversal—by a large and significant margin. They show that the image representation is a key driver of the model's success, as other time series neural network specifications have difficulty matching the image CNN's performance. Jiang

[11] Ch. 9 of Goodfellow *et al.* (2016) provide a more general introduction to CNN.

et al. (2022) note that their strategy may be partially approximated by replacing the CNN forecast with a simpler signal. In particular, a stock whose latest close price is near the bottom of its high-low range in recent days tends to appreciate in the subsequent week. This pattern, which is not previously studied in the literature, is detected from image data by the CNN and is stable over time and across market size segments and across countries.

Glaeser *et al.* (2018) study the housing market using images of residential real estate properties. They use a pre-trained CNN model (Resnet-101 from He *et al.*, 2016) to convert images into a feature vector for each property, which they then condense further using principal components, and finally the components are added to an otherwise standard hedonic house pricing model. Glaeser *et al.* (2018) find that property data derived from images improves out-of-sample fit of the hedonic model. Aubry *et al.* (2022) predict art auction prices via a neural network using artwork images accompanied by non-visual artwork characteristics, and use their model to document a number of informational inefficiencies in the art market. Obaid and Pukthuanthong (2022) apply CNN to classify photos on the *Wall Street Journal* and construct a daily investor sentiment index, which predicts market return reversals and trading volume. The association is strongest among stocks with more severe limits-to-arbitrage and during periods of elevated risk. They also find that photos convey alternative information to news text.[12]

[12] Relatedly, Deng *et al.* (2022) propose a theoretical model to rationalize how the graphical content of 10-K reports affects stock returns.

4

Risk-Return Tradeoffs

The previous section mainly focuses on supervised prediction models that do not take a stand on the risk-return tradeoff, and thus do not constitute asset pricing models. In this section, we develop factor pricing models using unsupervised and semi-supervised learning methods that model the risk-return tradeoff explicitly.

4.1 APT Foundations

The Arbitrage Pricing Theory (APT) of Ross (1976) lays the groundwork for data-driven, machine learning analysis of factor pricing models. It demonstrates that with a partial model specification—requiring in essence only a linear factor structure, a fixed number of factors, and the minimal economic assumption of no-arbitrage—we can learn the asset pricing model simply by studying factor portfolios and understanding which components of returns are diversifiable and which are not. In other words, the APT provides a blueprint for empirical analysis of the risk-return tradeoff without requiring any knowledge of the mechanisms that give rise to asset pricing factors. Machine learning methods for latent factor analysis can thus be leveraged to conduct new and powerful analyses of empirical asset pricing phenomena.

We also refer readers to a survey of return factor models by Giglio *et al.* (2022a). They organize the literature into categories based on whether factors are observable, betas are observable, or neither are observable. Observable factors or betas give rise to time series and cross section regression methodologies. In this survey, we focus on the more challenging case in which factors and betas are latent (or, at best, partially observable), which allows us to go deeper into some of the details of machine learning factor modeling techniques.

4.2 Unconditional Factor Models

The premise of Ross (1976)'s APT is the following statistical factor model:

$$R_t = \alpha + \beta F_t + \epsilon_t, \tag{4.1}$$

where R_t collects individual asset excess returns $R_{i,t}$ into an $N \times 1$ vector, β is an $N \times K$ matrix of loadings on the $K \times 1$ vector of latent factors, F_t, which may have a non-zero mean $\gamma = \mathrm{E}(F_t)$ interpreted as factor risk premia, and ϵ_t is an $N \times 1$ vector of mean-zero residuals.

The $N \times 1$ intercept vector α represents pricing errors. They are the components of expected asset returns that are not explained by factor exposures. Ross (1976)'s APT and its descendants (e.g. Huberman, 1982; Ingersoll, 1984; Chamberlain and Rothschild, 1983) establish that no-near-arbitrage is equivalent to[1]

$$\alpha' \mathrm{Var}(\epsilon_t)^{-1} \alpha \lesssim_{\mathrm{P}} 1, \quad \text{as} \quad N \to \infty. \tag{4.2}$$

That is, the compensation for bearing idiosyncratic risk does not explode as the investment universe expands.

4.2.1 Estimating Factors via PCA

Motivated by the APT, Chamberlain and Rothschild (1983), Connor and Korajczyk (1986), and Connor and Korajczyk (1988) advocate PCA as a factor model estimator when factors and betas are latent. A

[1] We use $a \lesssim_{\mathrm{P}} b$ to denote $a = O_{\mathrm{P}}(b)$, and use $a \asymp_{\mathrm{P}} b$ if $a \lesssim_{\mathrm{P}} b$ and $b \lesssim_{\mathrm{P}} a$.

more convenient yet equivalent approach is to conduct singular value decomposition (SVD) of the de-meaned returns $\bar{R} = R - (\frac{1}{T}\sum_{t=1}^{T} R_t)\iota_T'$:

$$\bar{R} = \sum_{j=1}^{\widehat{K}} \sigma_j \varsigma_j \xi_j' + \widehat{U}, \tag{4.3}$$

where $\{\sigma_j\}$, $\{\varsigma_j\}$, and $\{\xi_j\}$ correspond to the first $\widehat{K}$ singular values, left and right singular vectors of $\bar{R}$, $\widehat{K}$ is any consistent estimator (e.g. Bai and Ng, 2002) of the number of factors in R_t, $\widehat{U}$ is a matrix of residuals, and ι_T is a $T \times 1$ vector of ones. This decomposition yields estimates of factor innovations $V_t = F_t - \mathrm{E}(F_t)$ and risk exposures β as

$$\widehat{V} = T^{1/2}(\xi_1, \xi_2, \dots, \xi_{\widehat{K}})', \quad \widehat{\beta} = T^{-1/2}(\sigma_1\varsigma_1, \sigma_2\varsigma_2, \dots, \sigma_{\widehat{K}}\varsigma_{\widehat{K}}). \tag{4.4}$$

The factor estimates are normalized such that they satisfy $\widehat{V}\widehat{V}' = T\mathbb{I}_{\widehat{K}}$. Alternatively, we can normalize $\widehat{\beta}$ such that $\widehat{\beta}'\widehat{\beta} = N\mathbb{I}_{\widehat{K}}$. There is fundamental indeterminacy in latent factor models, i.e., rotating factors and counter-rotating loadings leads to no change in the data-generating process. Therefore, factors and their loadings are identifiable only up to an invertible linear transformation (i.e., rotation). In light of this, different normalizations yield equivalent estimators of factors and loadings, in that they only differ by some rotation. Bai (2003) proves the consistency of these PCA estimates and derives their asymptotic distributions under the assumption that all factors are pervasive, i.e., $\lambda_K(\beta'\beta) \asymp_{\mathrm{P}} N$.

Connor and Korajczyk (1988) first investigate the performance of a latent factor model using a large cross section of roughly 1,500 stocks. They find that, while a PCA-based factor model performs better than the CAPM in explaining the risk-return tradeoff in their sample, it admits large and significant pricing errors. Generally speaking, unconditional factor models have a difficult time describing stock-level data. Based on this and related studies, unconditional latent factor models (and their estimation via PCA) largely fell out of favor in the period following Connor and Korajczyk (1988). Kelly *et al.* (2020b) corroborate those findings in more recent data. They show that, in the panel of CRSP stocks from 1962–2014, PCA is extremely unreliable for describing risk premia of individual stocks.

There is a recent resurgence of PCA for return factor modeling. This emanates in large part from the fact that, while PCA suffers when describing individual stock panels, it has much greater success in modeling panels of portfolios. For example, Kelly *et al.* (2020b), Kozak *et al.* (2018), and Pukthuanthong *et al.* (2019) demonstrate that factor models estimated from a panel of anomaly portfolios manage to price those portfolios with economically small pricing errors. These analyses build on earlier work of Geweke and Zhou (1996) who use a Gibbs sampling approach to extract latent factors from portfolio-level data.

A potential drawback of the latent factor approach lies in its difficulty in interpreting the estimated factors due to the indeterminacy of latent factor models. Yet, it is beneficial to adopt the latent-factor approach whenever the object of interest is invariant to rotations. We provide one such example next.

4.2.2 Three-pass Estimator of Risk Premia

A factor risk premium describes the equilibrium compensation investors demand to hold risk associated with that factor. Many theoretical economic models are developed on the basis of some non-tradable factors (factors that are not themselves portfolios) such as consumption, GDP growth, inflation, liquidity, and climate risk. To estimate the risk premium of a non-tradable factor we need to construct a factor mimicking portfolio and estimate its expected returns. To fix ideas, suppose that this non-tradable factor, G_t, relates to the cross-section of assets in the following form:

$$G_t = \xi + \eta V_t + Z_t, \tag{4.5}$$

where Z_t is measurement error and $V_t = F_t - \mathrm{E}(F_t)$. According to this model, the risk premium of G_t is given by $\eta\gamma$. Since G_t is sometimes motivated from some economic theory, its mimicking portfolio and risk premium may be economically interpretable. Moreover, ηV_t and $\eta\gamma$ are rotation invariant and identifiable despite that V_t, η, and γ are identifiable only up to some rotation. To see this, it is known from Bai and Ng (2002) that there exists a matrix H such that $\widehat{V}_t \xrightarrow{P} HV_t$, for any t. If we rewrite the DGP of R_t and G_t with respect to HV_t, then the risk premium of V_t is $H\gamma$, and the loading of G_t on HV_t becomes ηH^{-1}, yet the innovation of the mimicking portfolio and the risk premium of G_t remain $\eta H^{-1} HV_t = \eta V_t$ and $\eta H^{-1} H\gamma = \eta\gamma$.

Giglio and Xiu (2021) propose a three-pass estimator to make inference on $\eta\gamma$, marrying Fama-MacBeth regression with PCA. The first pass is to conduct PCA and estimate factors and loadings according to (4.4). The second pass recovers the risk premia of latent factors via Fama-MacBeth regressions:

$$\widehat{\gamma} = \frac{1}{T}\sum_{t=1}^{T}(\widehat{\beta}'\widehat{\beta})^{-1}\widehat{\beta}' R_t. \tag{4.6}$$

The third pass recovers the loading of G_t on the estimated latent factors:

$$\widehat{\eta} = \frac{1}{T}\sum_{t=1}^{T} G_t \widehat{V}_t'.$$

The risk premium estimator thereby becomes: $\widehat{\eta}\widehat{\gamma}$.

Giglio and Xiu (2021) establish the asymptotic property of the resulting estimator. Their asymptotic analysis of Fama-MacBeth on PCA outputs paves the way for statistical inference on quantities of interest in asset pricing with latent factor models, including risk premia, stochastic discount factors (Giglio *et al.*, 2021b), and alphas (Giglio *et al.*, 2021a).

The three-pass estimator is closely related to PCA-regression based mimicking portfolios. The procedure amounts to regressing G_t onto the PCs of R_t to construct its factor-mimicking portfolio and computing its average return to arrive at G_t's risk premium. The use of PCs instead of the original assets in R_t is a form of regularization. This viewpoint encourages the adoption of other regularization methods in machine learning, like ridge and lasso, when creating mimicking portfolios.

On the empirical side, Table 4.1 collects risk premia estimates using different methods for a number of non-tradable factors, including AR(1) innovations in industrial production growth (IP), VAR(1) innovations in the first three principal components of 279 macro-finance variables from Ludvigson and Ng (2010), the liquidity factor of Pástor and Stambaugh (2003), the intermediary capital factor from He *et al.* (2017), four factors from Novy-Marx (2014) (high monthly temperature in

Manhattan, global land surface temperature anomaly, quasiperiodic Pacific Ocean temperature anomaly or "El Niño," and the number of sunspots), and an aggregate consumption-based factor from Malloy *et al.* (2009).

This table highlights two issues of the conventional two-pass regressions: the omitted variable bias and the measurement error bias. The two-pass estimates depend on the benchmark factors researchers select as controls. Yet economic theory often provides no guidance on what factors should serve as controls. Omitting control factors would in general bias the risk premia estimates. Take the liquidity and intermediary capital factors as an example. Risk premium estimates for the former change from 226bp per month based on a univariate two-pass regression to 57bp based on the same method but with FF3 factors as control. Similarly, estimates for the latter change from 101bp to 43bp.

Equation (4.5) also allows for noisy ($\eta = 0$) and weak (η small) factors as special cases, which normally would distort inference on risk premia in a two-pass regression (a phenomenon first documented by Kan and Zhang, 1999). For instance, the four factors from Novy-Marx (2014) are examples of variables that appear to predict returns in standard predictive regressions, but their economic link to the stock market seems weak. Nonetheless, three out of these four factors have significant risk premia based on two-pass regression with FF3 factors. Macro factors (such as PCs or consumption growth) are also weak. The three-pass approach addresses both omitted variable bias and measurement error because it estimates latent factors in the first pass, uses them as controls in the second-pass cross sectional regression, and adopts another time-series regression in the third pass to remove measurement error. Thanks to this robustness, the estimates from the last two columns of Table 4.1 appear more economically reasonable.

4.2.3 PCA Extensions

While PCA is the most common approach to factor recovery, there are alternatives with unique features. For instance, Giglio *et al.* (2021a) adopt matrix completion to estimate the factor model that can cope with an unbalanced panel of returns. Suppose that Ω is an $N \times T$ matrix

Factors	Two-pass w/o controls		Two-pass w/ FF3		Three-pass regression	
Liquidity	2.26^{**}	(0.90)	0.57	(0.68)	0.37^{**}	(0.16)
Interm. Cap.	1.01^{**}	(0.45)	0.43	(0.45)	0.60^{**}	(0.31)
NY temp.	-319.01	(255.73)	-277.96^{**}	(124.08)	-0.69	(13.90)
Global temp.	-6.65	(4.85)	-3.33	(2.07)	0.05	(0.21)
El Niño	56.85^{***}	(17.42)	-15.34^{**}	(7.11)	0.41	(0.82)
Sunspots	-409.37	(937.73)	882.89^{**}	(405.40)	4.01	(35.63)
IP growth	-0.36^{***}	(0.14)	-0.14^{**}	(0.05)	-0.01^{*}	(0.00)
Macro PC1	84.90^{***}	(24.76)	39.96^{***}	(13.57)	3.26^{**}	(1.58)
Macro PC2	9.35	(15.93)	23.91^{***}	(8.97)	-0.88	(1.27)
Macro PC3	-5.94	(14.30)	-31.24^{***}	(9.74)	-1.25	(1.51)
Cons. growth	0.26^{*}	(0.16)	0.07	(0.05)	0.00	(0.01)

Table 4.1: Three-Pass Regression: Empirical Results

Note: For each factor, the table reports estimates of risk premia in percentage points per month using different methods, with the restriction that the zero-beta rate is equal to the observed T-bill rate: two versions of the two-pass cross-sectional regression, using no control factors in the model and using the Fama-French three factors, respectively; the three-pass estimator using 7 latent factors. The test assets comprise of 647 portfolios sorted by various characteristics (from Table B1 of Giglio and Xiu, 2021).

with the (i, t) element equal to 1 if and only if $R_{i,t}$ is not missing. The matrix completion algorithm solves the following convex optimization problem:

$$\widehat{X} = \arg\min_{X} \|(R - X) \circ \Omega\|^2 + \lambda \|X\|_n,$$

where $\circ$ denotes Hadamard product, $\|X\|_n = \sum_{i=1}^{\min\{N,T\}} \psi_i(X)$ with $\psi_i(X)$ being the ith largest singular value of X, and λ is a tuning parameter. By penalizing the ℓ_1-norm of singular values of X, the algorithm attempts to find a low rank approximation of the return matrix R using only its observed entries. The estimated $\widehat{X}$ is a completed matrix of R with a low rank structure, from which latent factors and loadings can thus be recovered via SVD.

The standard implementation of PCA applies SVD to demeaned excess returns matrix $\bar{R}$. In doing so, it estimates latent factors and betas solely from the sample *centered* second moment of returns. Lettau and Pelger (2020b) point out that PCA's sole reliance on second moment information leads to inefficient factor model estimates. Asset pricing theory implies a restriction between asset means and factor betas (in particular, through the unconditional version of Euler equation (1.2)). They thus argue that leaning more heavily on factor loading information contained in the first moment of returns data can improve overall performance of PCA estimators. Based on this insight, they develop "risk-premia PCA" (RP-PCA) estimator, which applies PCA to an *uncentered* second moment of returns, $T^{-1}\sum_{t=1}^{T} R_t R_t' + \lambda(T^{-1}\sum_{t=1}^{T} R_t)(T^{-1}\sum_{t=1}^{T} R_t)'$, where $\lambda > -1$ serves as a tuning parameter. Connor and Korajczyk (1988) also use uncentered PCA, but stick to the case of $\lambda = 0$, whereas the standard PCA corresponds to $\lambda = -1$.

Lettau and Pelger (2020a) establish the asymptotic theory of RP-PCA and show that it is more efficient than PCA when factors are pervasive in the absence of pricing error. While the standard PCA approach is robust to the existence of pricing error, this error term could bias the RP-PCA estimator in that the expected returns are no longer parallel to factor loadings. We conjecture that such a bias is asymptotically negligible if the economic constraint of no-near-arbitrage (4.2) is imposed, in which case the magnitude of α is sufficiently small that it would not bias estimates of factors and their loadings asymptotically.[2]

Giglio *et al.* (2021b) point out that the strength of a factor hinges on the choice of test assets. Even the market factor could become weak if all test assets are long-short portfolios that have zero exposure to this factor. To resolve this weak factor issue in risk premia estimation, they propose a procedure to select test assets based on supervised PCA, as discussed in Section 3.5.2. Moreover, this procedure can be applied to detect missing factors in models of the stochastic discount factor.

4.2.4 Which Factors?

The search for factors that explain the cross section of expected stock returns has produced hundreds of potential candidates. Many of them are redundant, adding no explanatory power for asset pricing given some other factors. Some are outright useless that have no explanatory power.

Machine learning methods can address the challenge of redundant and useless factors through dimension reduction and variable selection.

[2] Bryzgalova *et al.* (2023) introduce other economically motivated targets to identify later pricing factors.

For example, a lasso regression of average returns on factor covariances can help identify a parsimonious set of factors that price the cross section of assets. At the same time, selection mistakes are inevitable: overfitting may lead to the selection of useless variables; variables that have relatively weak explanatory power may be omitted; redundant variables may be selected in place of the real ones. Figure 4.1 by Feng *et al.* (2020) show that the variables selected by lasso vary considerably across different random seeds adopted in cross validation, which randomly split the sample into multiple folds (see Figure 3.1).

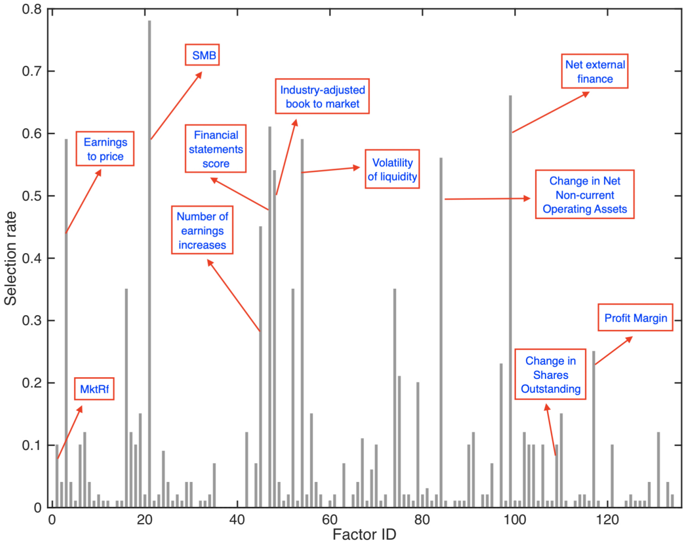


Figure 4.1: Factor Selection Rate

Note: Source: Feng *et al.* (2020). The figure shows, for each factor identified by the factor ID (on the X axis), in what fraction of the random seeds each factor is selected by a lasso regression of average returns onto factor covariances via 200 random cross validations.

Feng *et al.* (2020) propose a methodology that marries the double machine learning framework of Chernozhukov *et al.* (2018) with two-pass cross-sectional regression to identify a coherent set of factors and develop the asymptotic distribution of their estimator, with which they can make inferences regarding the factors.

Empirically, Feng *et al.* (2020) apply their inference procedure recursively to distinguish useful factors from useless and redundant factors as they are introduced in the literature. Their empirical findings show that had their approach been applied each year starting in 1994, only 17 factors out of 120+ candidate factors would have been considered useful, with a large majority identified as redundant or useless.

Another literature considers factor model selection from a model uncertainty and model averaging perspective, with related work including Avramov *et al.* (2021) and Chib *et al.* (2023). Avramov *et al.* (2021) shows that prior views about how large a Sharpe ratio could be has implications for the inclusion of both factors and predictors. In general, Bayesian model uncertainty is an interesting topic for further development in financial machine learning.

4.3 Conditional Factor Models

The previous section focuses on the unconditional version of Euler equation (1.2) with static betas and risk prices (derived by replacing $\mathcal{I}_t$ with the null set, or "conditioning down"). Generally, when we change the conditioning set, the factor representation also changes because assets' conditional moments change. It *could* be the case that as we condition down asset betas and expected returns do not change—in this special case the conditional and unconditional models are the same. However, empirical research demonstrates that asset covariances are highly predictable in the time series, and a preponderance of evidence suggests that asset means are also predictable. In other words, we can rule out the "conditionally static" special case.

To condition, or not to condition? That is the question when devising return factor models. Our view is that, whenever possible, the researcher should aspire to build an effective conditional model. Conditional models are ambitious—they describe the state dependent nature of asset prices, and thus capture in finer resolution the behavior of markets. However, conditional models are also more demanding, requiring the researcher to supply relevant data to summarize prevailing conditions. Such conditioning information may be expansive and may require more richly parameterized models in order to capture nuanced conditional behaviors. When the relevant conditioning information is unavailable, an unconditional model allows the researcher to understand basic asset behavior without necessarily having to understand detailed market dynamics and with potentially simpler models. For this reason, much of the early literature on return factor analysis pursues unconditional specifications (as in the preceding section). In this section we focus on conditional model formulations.

In analogy to (4.1), the vectorized *conditional* latent factor model is

$$R_{t+1} = \alpha_t + \beta_t F_{t+1} + \epsilon_{t+1}, \tag{4.7}$$

where factor loadings and pricing errors now vary with the conditioning information set $\mathcal{I}_t$.

4.3.1 IPCA

Without additional restrictions, the right-hand side of (4.7) contains too many degrees of freedom and the model cannot be identified. Instrumented principal components analysis (IPCA) of Kelly *et al.* (2020b) makes progress by introducing restrictions that link assets' betas (and alphas) to observables. The IPCA model takes the form

$$R_{t+1} = \underbrace{Z_t\Gamma_\alpha}_{\alpha_t} + \underbrace{Z_t\Gamma_\beta}_{\beta_t} F_{t+1} + \epsilon_{t+1}, \tag{4.8}$$

where Z_t is an $N \times L$ matrix that stacks up data on L observable characteristics (or "instruments") for each asset.[3] F_{t+1} is again a $K \times 1$ vector of latent factors. Harvey and Ferson (1999), and more recently Gagliardini *et al.* (2016), also model factor loadings as time-varying functions of observables, but their factors are fully observable.

The core of the IPCA model is its specification of β_t. First and foremost, time-varying instruments directly incorporate dynamics into conditional factor loadings. More fundamentally, incorporating instruments allows additional data to shape the factor model, differing from unconditional latent factor techniques like PCA that estimate the factor structure only from returns data. And anchoring loadings to observables

[3] Typically, one of the characteristics in Z is a constant term.

identifies the model by partially replacing unidentified parameters with data.

The $L \times K$ matrix Γ_β defines the mapping from a potentially large number of characteristics (L) to a small number of risk factor exposures (K). When we estimate Γ_β, we are searching for a few linear combinations of candidate characteristics that best describe the latent factor loading structure. To appreciate this point, first imagine a case in which Γ_β is the L-dimensional identity matrix. In this case, it is easy to show that F_t consists of L latent factors that are proportional to $R'_{t+1}Z_t$, and the characteristics determine the betas on these factors. This is reminiscent of Rosenberg (1974) and MSCI Barra (a factor risk model commonly used by practitioners). Barra's model includes several dozen characteristics and industry indicators in Z_t. When the number of firm characteristics L is large, the number of free parameters is equal to the number of factor realizations $\{F_t\}$, or $L \times T$, which is typically large compared to sample size. There is significant redundancy in Barra factors (a small number of principal components capture most of their joint variation) which suggests it may be overparameterized and inefficient.

IPCA addresses this problem with dimension reduction in the characteristic space. If there are many characteristics that provide noisy but informative signals about a stock's risk exposures, then aggregating characteristics into linear combinations isolates their signal and averages out their noise.

The challenge of migrating assets, such as stocks evolving from small to large or from growth to value, presents a problem in modeling stock-level conditional expected returns using simple time series methods. The standard solution is to create portfolios that possess average characteristic values that are somewhat stable over time, but this approach becomes impractical if multiple characteristics are needed to accurately describe an asset's identity. The IPCA solution parameterizes betas with the characteristics that determine a stock's risk and return. IPCA tracks the migration of an asset's identity through its betas, which are in turned defined by the asset's characteristics. This eliminates the need for a researcher to manually group assets into portfolios since the model explicitly tracks assets' identities in terms of their characteristics. As a

				K		
Test Assets	Statistic	1	3	4	5	6
				Panel A: IPCA		
Stocks	Total R^2	14.9	17.6	18.2	18.7	19
	Pred. R^2	0.36	0.43	0.43	0.70	0.70
	N_p	636	1908	2544	3180	3816
Portfolios	Total R^2	90.3	97.1	98.0	98.4	98.8
	Pred. R^2	2.01	2.10	2.13	2.41	2.39
	N_p	636	1908	2544	3180	3816
				Panel B: Observable Factors		
Stocks	Total R^2	11.9	18.9	20.9	21.9	23.7
	Pred. R^2	0.31	0.29	0.28	0.29	0.23
	N_p	11452	34356	45808	57260	68712
Portfolios	Total R^2	65.6	85.1	87.5	86.4	88.6
	Pred. R^2	1.67	2.07	1.98	2.06	1.96
	N_p	37	111	148	185	222

Table 4.2: IPCA Comparison with Other Factor Models

Note: The table reports total and predictive R^2 in percent and number of estimated parameters (N_p) for the restricted ($\Gamma_\alpha = \mathbf{0}$) IPCA model (Panel A) and for observable factor models with static loadings (Panel B). Observable factor model specifications are CAPM, FF3, FFC4, FF5, and FFC6 in the $K = 1, 3, 4, 5, 6$ columns, respectively (from Table 2 of Kelly *et al.*, 2020b).

result, the model accommodates a high-dimensional system of assets (e.g., individual stocks) without the need for ad hoc portfolio formation.

Finally, the IPCA specification in (4.7) also allows for the possibility that characteristics may proxy for alpha instead of beta. Conventional asset pricing models assume that differences in expected returns among assets are solely due to differences in risk exposure. However, when the $L \times 1$ coefficient vector Γ_α is non-zero, stock-level characteristics can predict returns in a way that does not align with the risk-return tradeoff. IPCA addresses this by estimating alpha as a linear combination of characteristics (determined by Γ_α) that best explains conditional expected returns while controlling for the characteristics' role in factor risk exposures. If the alignment of characteristics with average stock returns differs from the alignment of characteristics with risk factor loadings, IPCA will estimate a non-zero Γ_α, thus identifying mispricing (that is, compensation for holding assets that is unrelated to the assets' systematic risk exposure).

Table 4.2 compares IPCA (Panel A) with various numbers of latent factors to other leading models in the literature. The first set of comparison models includes pre-specified observable factors, estimated using the traditional approach of asset-by-asset time series regression. The $K = 1$ model is the CAPM, $K = 3$ is the Fama-French (1993) three-factor model that includes the market, SMB and HML ("FF3" henceforth), $K = 4$ is the Carhart (1997, "FFC4") model that adds MOM to the FF3 model, $K = 5$ is the Fama-French (2015, "FF5") five-factor model that adds RMW and CMA to the FF3 factors, $K = 6$ ("FFC6") includes MOM alongside the FF5 factors. All models in Table 4.2 are estimated with a zero-intercept restriction by imposing $\Gamma_\alpha = \mathbf{0}$ in IPCA or by omitting an intercept in time series regressions.

Table 4.2 reports the total R^2 based on contemporaneous factor realizations, the predictive R^2 (replacing the factor realization with its average risk premium), as well as the number of estimated parameters (N_p) for each model. We report fits at the individual stock level and at the level of characteristic-managed portfolios. These statistics are calculated in-sample.[4] For individual stocks, observable factor models produce a slightly higher total R^2 than IPCA. To accomplish this, however, observable factors rely on vastly more parameters than IPCA. In this sample of 11,452 stocks with 37 instruments over 599 months, observable factor models estimate 18 times ($\approx 11452/(37 + 599)$) as many parameters as IPCA! In short, IPCA provides a similar description of systematic risk in stock returns as leading observable factors while using almost 95% fewer parameters. At the same time, IPCA provides a substantially more accurate description of stocks' risk compensation than observable factor models, as evidenced by the predictive R^2. For characteristic-managed portfolios, observable factor models' total R^2 and predictive R^2 both suffer in comparison to IPCA.

Figure 4.2 compares models in terms of average pricing errors for 37 characteristic-managed "anomaly" portfolios.[5] The left-hand plot shows portfolio alphas from the FFC6 model against their raw average

[4] Kelly *et al.* (2020b) show that IPCA is highly robust out-of-sample, while other models like those using observable factors or PCA tend to suffer more severe out-of-sample deterioration.

[5] For comparability, portfolios are re-signed and scaled to have positive means and 10% annualized volatility.

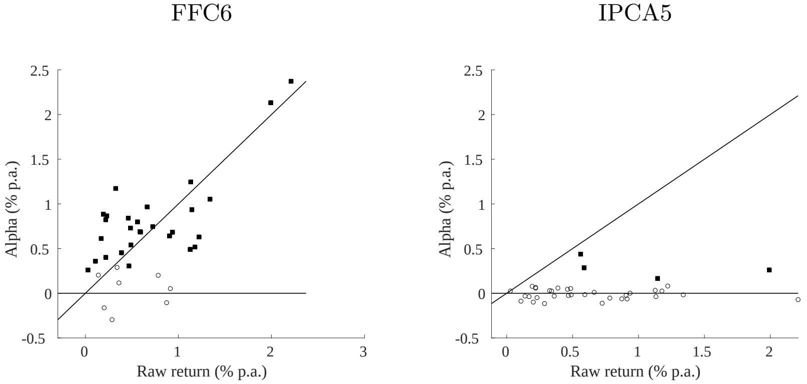


Figure 4.2: Alphas of Characteristic-Managed Portfolios

Note: The left and right panels report alphas for characteristic-managed portfolios relative to the FFC6 model and the IPCA five-factor model. Alphas are plotted against portfolios' raw average excess returns. Alphas with t-statistics in excess of 2.0 are shown with filled squares, while insignificant alphas are shown with unfilled circles (from Figure 1 of Kelly *et al.*, 2020b).

excess returns, overlaying the 45-degree line. Alphas with t-statistics in excess of 2.0 are depicted with filled squares, while insignificant alphas are shown with unfilled circles. Twenty-nine characteristic-managed portfolios earn significant alpha with respect to the FFC6 model. The alphas are clustered around the 45-degree line, indicating that their average returns are essentially unexplained by observable factors. The right-hand plot shows the time series averages of conditional alphas from the five-factor IPCA specification. Four portfolios have conditional alphas that are significantly different from zero but are economically small. Figure 4.2 supports the conclusion that the IPCA latent factor formulation is more successful at pricing a range of equity portfolios relative to asset pricing models with observable factors.

The IPCA framework has been used to study cross-sectional asset pricing in a variety of markets, including international stocks (Langlois, 2021; Windmueller, 2022), corporate bonds (Kelly *et al.*, Forthcoming), equity index options (Büchner and Kelly, 2022), single-name equity options (Goyal and Saretto, 2022), and currencies (Bybee *et al.*, 2023a). It has also been used to understand the profits of price trend signals

(Kelly *et al.*, 2021) and the narrative underpinnings of asset pricing models (Bybee *et al.*, 2023b).

4.4 Complex Factor Models

A number of papers study generalizations of specification (4.8) for instrumented betas in latent conditional factor models. IPCA can be viewed as a linear approximation for risk exposures based on observable characteristics data. While many asset pricing models predict nonlinear association between expected returns and state variables, the theoretical literature offers little guidance for winnowing the list of conditioning variables and functional forms. The advent of machine learning allows us to target this functional form ambiguity with a quiver of nonlinear models.

In early work, Connor *et al.* (2012) and Fan *et al.* (2016b) allow for nonlinear beta specification by treating betas as nonparametric functions of conditioning characteristics (but, unlike IPCA, the characteristic are fixed over time for tractability). Kim *et al.* (2020) adopt this framework to study the behavior of "arbitrage" portfolios that hedge out factor risk.

Gu *et al.* (2020a) extends the IPCA model by allowing betas to be a neural network function of characteristics. Figure 4.3 diagrams their "conditional autoencoder" (CA) model. Figure 4.3 illustrates its basic structure, which differs from (4.8) by propagating the input data (instruments Z_t) through nonlinear activation functions. The CA is the first deep learning model of equity returns that explicitly accounts for the risk-return tradeoff. Gu *et al.* (2020a) show that CA's total R^2 is similar to that of IPCA, but it substantially improves over IPCA in terms of predictive R^2. In other words, the CA provides a more accurate description of assets' conditional compensation for factor risk.

Gu *et al.* (2020a) is a high complexity model and its strong empirical performance hints at a benefit of complexity in factor models analogous to that studied in prediction models by Kelly *et al.* (2022a). Didisheim *et al.* (2023) formalize this idea and prove the virtue of complexity in factor pricing. They build their analysis around the conditional stochastic discount factor (SDF), which can be generally written as a portfolio of

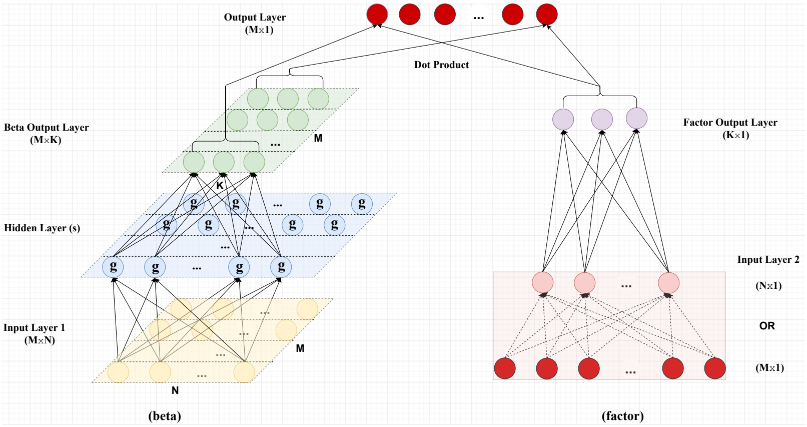


Figure 4.3: Conditional Autoencoder Model

Note: This figure presents the diagram of a conditional autoencoder model, in which an autoencoder is augmented to incorporate covariates in the factor loading specification. The left-hand side describes how factor loadings (in green) depend on firm characteristics (in yellow) of the input layer 1 through an activation function g on neurons of the hidden layer. Each row of yellow neurons represents the vector of characteristics of one ticker. The right-hand side describes the corresponding factors. Nodes (in purple) are weighted combinations of neurons of the input layer 2, which can either be characteristic-managed portfolios (in pink) or individual asset returns (in red). In the former case, we further use dashed arrows to indicate that the characteristic-managed portfolios rely on individual assets through pre-determined weights (not to be estimated). In either case, the effective input can be regarded as individual asset returns, exactly what the output layer (in red) aims to approximate, thus this model shares the same spirit as a standard autoencoder. Source: Gu *et al.* (2020a).

risky assets:

$$M_{t+1} = 1 - w(X_t)' R_{t+1}, \tag{4.9}$$

where R_{t+1} is the vector of excess returns on the N assets. The N-vector $w(X_t)$ contains the SDF's conditional portfolio weights, with X_t representing conditioning variables that span the time t information set. Absent knowledge of its functional form, $w()$ can be approximated with a machine learning model:

$$w(X_t) \approx \sum_{p=1}^{P} \lambda_p S_p(X_t)$$

where $S_p(X_t)$ is some nonlinear basis function of X_t and the number of parameters P in the approximation is large. A machine learning model of the SDF may be interpreted as a factor pricing model with P factors:

$$M_{t+1} \approx 1 - \sum_p \lambda_p F_{p,t+1}, \tag{4.10}$$

where each "factor" $F_{p,t+1}$ is a managed portfolio of risky assets using the nonlinear asset "characteristics" $S_p(X_t)$ as weights. The main theoretical result of Didisheim *et al.* (2023) shows that quite the more factors in an asset pricing model the better. In this setup, adding factors means using a richer representation of the information contained in X_t which achieves a better approximation of the true SDF. The improvement in approximation accuracy dominates statistical costs of having to estimate many parameters. As a result, expected out-of-sample alphas decrease as the number of factors grows. This interpretation of the virtue of complexity is a challenge for the traditional APT perspective that a small number of risk factors provide a complete description of the risk-return tradeoff for any tradable assets. This has the implication that, even if arbitrage is absent and an SDF exists, the fact that the SDF must be estimated implies that it is possible (in fact, expected) to continually find new empirical "risk" factors that are unpriced by others and that adding these factors to the pricing model continually improves its out-of-sample performance.

4.5 High-frequency Models

The increasing availability of high-frequency transaction-level data on a growing cross-section of tradable assets presents a unique opportunity in estimating risks of individual assets and their interdependencies. Simple nonparametric measures of volatility and covariances (Andersen and Bollerslev, 1998; Andersen *et al.*, 2001; Barndorff-Nielsen and Shephard, 2002) provide an early demonstration of how to harness rich and timely intraday prices data to better understand asset market fluctuations. The use of high frequency measures helps resolve several challenges of studying low frequency time series. For example, it helps the researcher accommodate structural breaks and time-varying parameters with minimal assumptions. Moreover, many standard assumptions on linearity, stationarity, dependence, and heteroscedasticity in classic time series are often unnecessary for modeling intraday data.

We identify two streams of recent literature that adopt machine learning techniques to estimate high-dimensional covariances and to improve volatility forecasting with high frequency data.

Accurate covariance estimates are critical to successful portfolio construction. But estimating large covariance matrices is a challenging statistical problem due to the curse of dimensionality. A number of methods rely on various forms of regularization (Bickel and Levina, 2008a; Bickel and Levina, 2008b; Cai and Liu, 2011; Ledoit and Wolf, 2012; Ledoit and Wolf, 2004) to improve estimates. Inspired by the APT, Fan *et al.* (2008) propose factor-model based covariance matrix estimators in the case of a strict factor model with observable factors, and Fan *et al.* (2013) offer an approach with an approximate factor structure with latent factors.

A factor structure is also necessary at high frequency when the dimension of the panel approaches the sample size. However, the econometric techniques are fundamentally different in low-frequency and high-frequency sampling environments. The latter is often cast in a continuous-time setting based on a general continuous-time semimartingale model, allowing for stochastic variation and jumps in return dynamics. Ait-Sahalia and Xiu (2019) develop the asymptotic theory of nonparametric PCA to handle intraday data, which paves the way for applications of factor models in continuous time. In addition, Fan *et al.* (2016a) and Ait-Sahalia and Xiu (2017) develop estimators of large covariance matrices using high frequency data for individual stocks on the basis of a continuous-time factor model.

A promising agenda is to tie the literature on high-frequency risk measurement with the literature on the cross-section of expected returns, leveraging richer risk information for a better understanding of the risk-return tradeoff. Some relevant research in this direction includes Bollerslev *et al.* (2016), who compute individual stock betas with respect to the continuous and the jump components of a single market factor in a continuous-time setting, but associate these estimates with the cross-section of returns in a discrete-time setup. Ait-Sahalia *et al.* (2021)

provide inference for the risk premia in a unified continuous-time framework, while also allowing for multiple factors and stochastic betas in the first stage, and treat the betas in the second pass as components that were estimated in the first pass, generalizing the classical approach to inference by Shanken (1992a). Empirically, they examine factor models of intraday returns using Fama-French and momentum factors sampled every 15 minutes built by Ait-Sahalia *et al.* (2020).

The idea of measuring volatility using high-frequency data also spurred a promising agenda in volatility forecasting. The heterogenous autoregressive (HAR) model of past realized volatility measures (Corsi, 2009) has emerged as the leading volatility forecasting model in academic research and industry practice. A number of recent papers have examined machine learning strategies for volatility forecasting, including Li and Tang (2022) and Bollerslev *et al.* (2022). But unlike a return prediction analysis in which machine learning predictions directly translate into higher Sharpe ratios, it is unclear the extent to which machine learning forecasts outperform existing HAR models in economic terms. This is an interesting open question in the literature.

4.6 Alphas

This section discusses the literature on alpha testing and machine learning. Alpha is the portion of the expected return unaccounted for by factor betas and is thus a model-dependent object. Because economic theory is often too stylized to pin down the identities of all factors, and because the data may not be rich enough to infer the true factors in a data-driven way, model misspecification is lingering challenge for distinguishing alpha from “fair” compensation for factor risk exposure. For example, it is possible that estimated alphas are a manifestation of weak factor betas, reminiscent of the omitted variable problem in regression. In other words, one person’s alpha is another’s beta. In a latent factor model, alphas and betas are ultimately distinguished by a factor strength cutoff that distinguishes factors from idiosyncratic noise.

We concentrate our alpha analysis from the perspective of unconditional latent factor models. Our emphasis on unconditional rather than conditional alphas is motivated by the focus in the literature. Our emphasis on latent factor models is motivated by our view that misspecification concerns are less severe for latent factor models.

4.6.1 Alpha Tests and Economic Importance

A longtime focus of empirical asset pricing is the null hypothesis that all alphas are equal to zero, $\mathbb{H}_0 : \alpha_1 = \alpha_2 = \ldots = \alpha_N = 0$. This is a single hypothesis distinct from a multiple hypotheses alpha testing problem that we discuss later. Rejection of $\mathbb{H}_0$ is interpreted as evidence of misspecification of the asset pricing model or mispricing of the test assets (and, perhaps mistakenly, as a violation of Ross (1976)'s APT).

The well known GRS test (Gibbons *et al.*, 1989) of $\mathbb{H}_0$ is a Chi-squared test designed for low-dimensional factor models with observable factors. Fan *et al.* (2015) and Pesaran and Yamagata (2017) propose tests of the same null but in high-dimensional settings. This is an important step forward as it eliminates a restriction ($T > N + K$) in the original GRS test and improves the power of the test when N is large. While these methods are initially proposed for models with observable factors, it is possible to extend them to latent factor models. In fact, Giglio *et al.* (2021a) construct an estimator of α using the estimated factor loadings and risk premia given by (4.4) and (4.6), respectively:

$$\widehat{\alpha} = \frac{1}{T}\sum_{t=1}^{T} R_t - \widehat{\beta}\widehat{\gamma}. \tag{4.11}$$

They also derive the necessary asymptotic expansion of $\widehat{\alpha}$, which paves the way for constructing tests of alpha in latent factor models.

The GRS test statistic is built upon $(S^{\star})^2 = \alpha'\Sigma_{\epsilon}^{-1}\alpha$, which can be interpreted as the optimal squared Sharpe ratio of a portfolio that has zero exposure to factors. Estimating this Sharpe ratio is one thing, implementing a trading strategy to realize it is another. That is, a rejection of the zero alpha hypothesis in a GRS-like test does not necessarily mean the rejection is economically important. Any meaningful quantification of the economic importance of alphas should consider the feasibility of a would-be arbitrageur's trading strategy. Evaluating statistical rejections in economic terms is both more valuable for asset pricing research and more relevant for practitioners.[6]

The APT assumes that arbitrageurs know the true parameters in the return generating process. Such an assumption might be benign provided a sufficiently large sample size, in which case the parameters are asymptotically revealed and arbitrageurs behave (approximately) as if they know the true parameters. The catch is that, in the setting of the APT, arbitrageurs must know an increasing number of alphas, thus the cross-section dimension is large relative to a typical sample size. Consequently, it is unreasonable to assume that arbitrageurs can learn alphas even in the large T limit.

Da *et al.* (2022) revisit the APT and relax the assumption of known parameters. In their setting, arbitrageurs must use a feasible trading strategy that relies on historical data with a sample size T. For any feasible strategy $\widehat{w}$ at time t, they define this strategy's next period conditional Sharpe ratio as:

$$S(\widehat{w}) := \mathrm{E}(\widehat{w}' R_{t+1} | \mathcal{I}_t) / \mathrm{Var}(\widehat{w}' R_{t+1} | \mathcal{I}_t)^{1/2},$$

where $\mathcal{I}_t$ is the information set at t. They show that $S(\widehat{w})$ obeys

$$S(\widehat{w}) \leq \left(S(\mathcal{G})^2 + \gamma' \Sigma_v^{-1} \gamma\right)^{1/2} + o_{\mathrm{P}}(1), \tag{4.12}$$

where $S(\mathcal{G})^2 := \mathrm{E}(\alpha|\mathcal{G})' \Sigma_\epsilon^{-1} \mathrm{E}(\alpha|\mathcal{G})$, as $N \to \infty$, Σ_v is the covariance matrix of factors, and $\mathcal{G}$ is the information set generated by $\{(R_s, \beta, V_s, \Sigma_\epsilon) : t - T + 1 \leq s \leq t\}$.

Notably, $\gamma' \Sigma_v^{-1} \gamma$ is the Sharpe ratio of the optimal factor portfolio. Therefore, for any factor-neutral strategy $\widehat{w}$, i.e., $\widehat{w}' \beta = 0$,

$$S(\widehat{w}) \leq S(\mathcal{G}) + o_{\mathrm{P}}(1). \tag{4.13}$$

This result suggests that it is the posterior estimate of α that determines the optimal feasible Sharpe ratio and imposes an upper bound on the

[6] As Shanken (1992b) puts

> *... practical content is given to the notion of 'approximate arbitrage,' by characterizing the investment opportunities that are available as a consequence of the observed expected return deviation ... Far more will be learned, I believe, by examining the extent to which we can approximate an arbitrage with existing assets.*

profits of statistical arbitrage. Any machine learning strategy, simple or complex, needs obey this feasible Sharpe ratio bound.

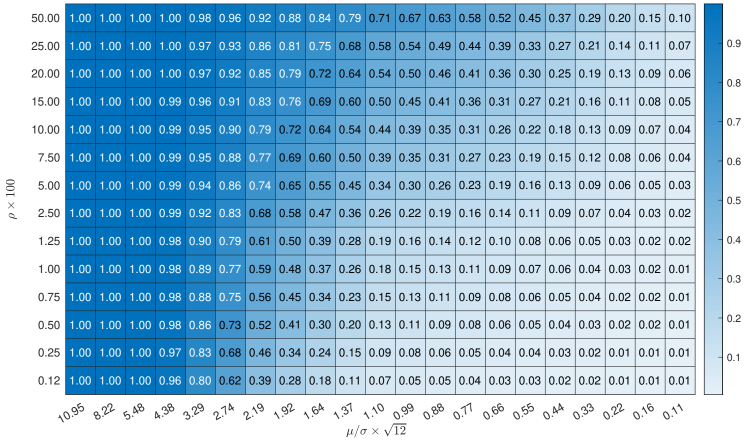
In general, $\mathrm{E}(S(\mathcal{G})^2) \leq \mathrm{E}((S^\star)^2)$, where the equality holds only when $\mathrm{E}(\alpha|\mathcal{G}) = \alpha$ almost surely. $S(\mathcal{G})$ results from the feasible strategy and $S^\star$ can be referred to as the infeasible optimal Sharpe ratio.

The "Sharpe ratio gap" between feasible and infeasible strategies characterizes the difficulty of statistical learning. If learning is hard, the gap is large. Figure 4.4 reports the ratio between these two Sharpe ratios numerically for a hypothetical return generating process in which α takes a value of zero with probability $1 - \rho$ and a value of μ or $-\mu$ each with probability $\rho/2$. Residuals have a covariance matrix Σ_ϵ that is a diagonal with variances of σ^2. Within this class of DGPs, μ/σ characterizes the strength of α (relative to noise), while ρ characterizes its rareness. Varying μ/σ and ρ helps trace out the role of various generating processes in arbitrageur performance, shown in Figure 4.4. As μ/σ increases, alpha is sufficiently strong and easy to learn, and the Sharpe ratio gap is smaller. Rareness of alpha plays a less prominent role, though more ubiquitous alpha also lead to a smaller gap.

Da *et al.* (2022) show how to quantify the gap between infeasible and feasible Sharpe ratios. They evaluate the APT on the basis of a 27-factor model, in which 16 characteristics and 11 GICS sector dummies are adopted as observable betas. The estimated infeasible Sharpe ratio is over 2.5 for a test sample from January 1975 to December 2020, and this is four times higher than the feasible Sharpe ratio of around 0.5 achieved by machine learning strategies. The fact that the feasible Sharpe ratio (before transaction costs) is below 0.5 suggests that the APT in fact works quite well empirically. In theory, the gap between feasible and infeasible Sharpe ratios further increases if arbitrageurs face more statistical obstacles, such as model misspecification, non-sparse residual covariance matrix, and so forth.

4.6.2 Multiple Testing

Since the introduction of CAPM, the financial economics community has collectively searched for "anomalies;" i.e., portfolios with alpha to the CAPM. Some of these, like size, value, and a handful of others, have



$\rho \times 100$	10.95	8.22	5.48	4.38	3.29	2.74	2.19	1.92	1.64	1.37	1.10	0.99	0.88	0.77	0.66	0.55	0.44	0.33	0.22	0.16	0.11
50.00	1.00	1.00	1.00	1.00	0.98	0.96	0.92	0.88	0.84	0.79	0.71	0.67	0.63	0.58	0.52	0.45	0.37	0.29	0.20	0.15	0.10
25.00	1.00	1.00	1.00	1.00	0.97	0.93	0.86	0.81	0.75	0.68	0.58	0.54	0.49	0.44	0.39	0.33	0.27	0.21	0.14	0.11	0.07
20.00	1.00	1.00	1.00	1.00	0.97	0.92	0.85	0.79	0.72	0.64	0.54	0.50	0.46	0.41	0.36	0.30	0.25	0.19	0.13	0.09	0.06
15.00	1.00	1.00	1.00	0.99	0.96	0.91	0.83	0.76	0.69	0.60	0.50	0.45	0.41	0.36	0.31	0.27	0.21	0.16	0.11	0.08	0.05
10.00	1.00	1.00	1.00	0.99	0.95	0.90	0.79	0.72	0.64	0.54	0.44	0.39	0.35	0.31	0.26	0.22	0.18	0.13	0.09	0.07	0.04
7.50	1.00	1.00	1.00	0.99	0.95	0.88	0.77	0.69	0.60	0.50	0.39	0.35	0.31	0.27	0.23	0.19	0.15	0.12	0.08	0.06	0.04
5.00	1.00	1.00	1.00	0.99	0.94	0.86	0.74	0.65	0.55	0.45	0.34	0.30	0.26	0.23	0.19	0.16	0.13	0.09	0.06	0.05	0.03
2.50	1.00	1.00	1.00	0.99	0.92	0.83	0.68	0.58	0.47	0.36	0.26	0.22	0.19	0.16	0.14	0.11	0.09	0.07	0.04	0.03	0.02
1.25	1.00	1.00	1.00	0.98	0.90	0.79	0.61	0.50	0.39	0.28	0.19	0.16	0.14	0.12	0.10	0.08	0.06	0.05	0.03	0.02	0.02
1.00	1.00	1.00	1.00	0.98	0.89	0.77	0.59	0.48	0.37	0.26	0.18	0.15	0.13	0.11	0.09	0.07	0.06	0.04	0.03	0.02	0.01
0.75	1.00	1.00	1.00	0.98	0.88	0.75	0.56	0.45	0.34	0.23	0.15	0.13	0.11	0.09	0.08	0.06	0.05	0.04	0.02	0.02	0.01
0.50	1.00	1.00	1.00	0.98	0.86	0.73	0.52	0.41	0.30	0.20	0.13	0.11	0.09	0.08	0.06	0.05	0.04	0.03	0.02	0.02	0.01
0.25	1.00	1.00	1.00	0.97	0.83	0.68	0.46	0.34	0.24	0.15	0.09	0.08	0.06	0.05	0.04	0.04	0.03	0.02	0.01	0.01	0.01
0.12	1.00	1.00	1.00	0.96	0.80	0.62	0.39	0.28	0.18	0.11	0.07	0.05	0.05	0.04	0.03	0.03	0.02	0.02	0.01	0.01	0.01

Figure 4.4: Ratios between $S(\mathcal{G})$ and $S^\star$

Note: The figure reports the ratios of optimal Sharpe ratios between feasible and infeasible arbitrage portfolios. The simulation setting is based on a simple model, in which only a $100 \times \rho\%$ of assets have non-zero alphas, with each entry corresponding to an annualized Sharpe ratio $\mu/\sigma \times \sqrt{12}$. Source: Da *et al.* (2022).

been absorbed into benchmark models (Fama and French, 1993; Fama and French, 2015). New anomalies are then proposed when researchers detect an alpha with respect to the prevailing benchmark. Harvey *et al.* (2016) survey the literature and collates a list of over three hundred documented anomalies. They raise the important criticism that the anomaly search effort has failed to properly account for multiple hypothesis testing when evaluating the significance of new anomalies.[7]

Multiple testing in this context refers to simultaneously testing a collection of null hypotheses: $\mathbb{H}_0^i : \alpha_i = 0$, for $i = 1, 2, \ldots, N$. This problem is fundamentally different from testing the single null hypothesis $\mathbb{H}_0 : \alpha_1 = \alpha_2 = \ldots = \alpha_N = 0$ discussed earlier. Multiple testing is prone to a false discovery problem because a fraction of the individual alpha tests are bound to appear significant due to chance alone and hence

[7] More broadly, Harvey (2017) among others highlight the propensity for finance researchers to draw flawed conclusions from their analyses by failing to account for unreported tests, failure to account for multiple tests, and career incentives that promote “p-hacking.”

their null hypotheses are incorrectly rejected.

Let t_i be a test statistic for the null $\mathbb{H}_0^i$. Suppose $\mathbb{H}_0^i$ is rejected whenever $|t_i| > c_i$ for some pre-specified critical value c_i. Let $\mathcal{H}_0 \subset \{1, ..., N\}$ denote the set of indices for which the corresponding null hypotheses are true. In addition, let $\mathcal{R}$ be the total number of rejections in a sample, and let $\mathcal{F}$ be the number of false rejections in that sample:

$$\mathcal{F} = \sum_{i=1}^{N} 1\{i \leq N : |t_i| > c_i \text{ and } i \in \mathcal{H}_0\}, \quad \mathcal{R} = \sum_{i=1}^{N} 1\{i \leq N : |t_i| > c_i\}.$$

Both $\mathcal{F}$ and $\mathcal{R}$ are random variables, but $\mathcal{R}$ is observable whereas $\mathcal{F}$ is not.

For any predetermined level $\tau \in (0,1)$, say, 5%, the individual tests ensure that the per-test error rate is bounded below by τ: $\mathbb{E}(\mathcal{F})/N \leq \tau$. In other words, the expected number of false rejections can be as large as $N\tau$. To curb the number of false rejections, an alternative proposal is to select a larger critical value to control the family-wise error rate (FWER): $\mathbb{P}(\mathcal{F} \geq 1) \leq \tau$. The latter proposal is unfortunately overly conservative in practice. The third proposal, dating back to Benjamini and Hochberg (1995), is to control the false discovery rate directly: FDR $\leq \tau$, where the false discovery proportion (FDP) and its expectation, FDR, are defined as FDP $= \mathcal{F}/\max\{\mathcal{R}, 1\}$ and FDR $=$ E(FDP).

While the asset pricing literature has long been aware of the general data snooping problem (Lo and MacKinlay, 1990; Sullivan *et al.*, 1999), early proposals suggest alternative single null hypotheses instead, such as $\mathbb{H}_0 : \max_i \alpha_i \leq 0$ or $\mathbb{H}_0 : \mathrm{E}(\alpha_i) = 0$ (see e.g. White, 2000; Kosowski *et al.*, 2006; Fama and French, 2010). Barras *et al.* (2010), Bajgrowicz and Scaillet (2012), and Harvey *et al.* (2016) are among the first to adopt FDR or FWER control methods in asset pricing contexts to curb multiple testing. Harvey and Liu (2020) propose a double-bootstrap method to control FDR, while also accounting for the false negative rate and odds ratio.[8] Giglio *et al.* (2021a) propose a rigorous inference approach to FDR control on alphas in a latent factor model, simultaneously tackling omitted variable bias, missing

[8] Chen *et al.* (2023) study hundreds of factors and find only 2 independent anomaly factors after controlling the FDR.

data, and high dimensionality in the test count. Jensen *et al.* (2021) propose a Bayesian hierarchical model to accomplish their multiple testing correction, which leverages a zero-alpha prior and joint behavior of factors, allowing factors' alpha estimates to shrink towards the prior and borrow strength from one another.

Ultimately, multiple testing is a statistical problem in nature. The aforementioned statistical methods typically meet the criteria of a good statistical test, such as controlling Type I error, false discovery rate, etc. Nonetheless, it is the economic performance that agents care about most. These two objective are typically in conflict. Jensen *et al.* (2021) and Da *et al.* (2022) point that multiple testing as a device for alpha selection often leads to extremely conservative trading strategies, even though it guards against FDR perfectly. Jensen *et al.* (2021) demonstrate that a researcher that includes factors in their portfolio based on a Bayesian hierarchical multiple testing approach would have achieved a large and significant improvement over an investor using the more conservative approach of FDR control.

5

Optimal Portfolios

In this section, we discuss and analyze machine learning approaches to portfolio choice. The portfolio choice problem lies at the heart of finance. It aims for efficient allocation of investor resources to achieve growth-optimal savings, and all major problems in asset pricing have an intimate association with it. Under weak economic assumptions (such as the absence of arbitrage), the mean-variance efficient portfolio (MVE) in an economy is a tradable representation of the stochastic discount factor (SDF), and thus summarizes the way market participants trade off risk and return to arrive at equilibrium prices (Hansen and Richard, 1987). Likewise, individual assets' exposures to the MVE portfolio map one-for-one into assets expected returns, meaning that the MVE amounts to a one-factor beta pricing model that explains cross-sectional differences in average returns (Roll, 1977). And, in the return prediction problem analyzed in Section 3, prediction performance is ubiquitously evaluated in terms of the benefits it confers in optimal portfolios.

There are many statistical approaches available for pursuing optimal portfolios. All approaches combine aspects of the distributional properties of assets (their risk and reward) and investors' preferences over the risk-return tradeoff. The seminal Markowitz (1952) problem endows investors with knowledge of the return distribution.[1] Without the need to estimate this distribution, portfolio choice is a one-step problem of utility maximization. Invoking Hayek's quote from the introduction, "*if we command complete knowledge of available means, the problem which remains is purely one of logic.*" This, once again, is emphatically *not* the economic problem the investor faces. The investor's portfolio choice problem in inextricably tied to estimation, which is necessary to confront their lack of knowledge about the return distribution. The investor makes portfolio decisions with highly imperfect information.

Since Markowitz, a massive literature proposes methodological tools to solve the estimation problem and arrive at beneficial portfolios. It is a stubborn problem whose nuances often thwart clever methodological innovations. Practical success hinges on an ability to inform the utility objective with limited data. In other words, the portfolio problem is ripe for machine learning solutions.

An emergent literature proposes new solutions rooted in machine learning ideas of shrinkage, model selection, and flexible parameterization. First, machine learning forecasts, derived from purely statistical considerations (such as return or risk prediction models) may be plugged into the portfolio choice problem and treated as known quantities. An investor that treats the estimation and portfolio objectives in isolation, however, is behaving suboptimally. Estimation noise acts as a source of risk for the investor's portfolio, so a complete segregation of the estimation problem unnecessarily sacrifices utility. Intuitively, the investor can do better by understanding the properties of estimation noise and accounting for it in their risk-return tradeoff calculations. This calls for integrated consideration of estimation and utility optimization, leading once again to problem formulations that are attractively addressed with machine learning methods. We begin our discussion by illustrating the limitations of naive solutions that segregate the estimation and utility maximization problems. This provides a foundational understanding

[1] There is a substantial theoretical literature that extends the Markowitz problem with sophisticated forms of preferences, return distributions, portfolio constraints, and market frictions, while maintaining the assumption that investors have all necessary knowledge of the return distribution. Brandt (2010) and Fabozzi *et al.* (2010) survey portions of this literature.

upon which we build our discussion of machine learning tools that improve portfolio choice solutions.

5.1 “Plug-in” Portfolios

We begin with an illuminating derivation from Kan and Zhou (2007) who analyze the impact of parameter estimation in a tractable portfolio choice problem. The one period return on N assets in excess of the risk free rate, R_t, is an i.i.d. normal vector with first and second moments $\mu \in \mathbb{R}^N$ and $\Sigma \in \mathbb{R}^{N\times N}$. Investor attitudes are summarized by quadratic utility with risk aversion γ, and the investor seeks a portfolio of risky assets $w \in \mathbb{R}^N$ (in combination with the risk-free asset) to maximize one-period expected utility:

$$\mathrm{E}[U(w)|\mu, \Sigma] = w'\mu - \frac{\gamma}{2}w'\Sigma w. \tag{5.1}$$

When the distribution of returns is known, the utility maximizing portfolio solution is

$$w^* = \frac{1}{\gamma}\Sigma^{-1}\mu. \tag{5.2}$$

Once we account for the reality of imperfect knowledge of the return distribution, investors must decide how to use available information in order to optimize the utility they derive from their portfolio. We assume that investors have at their disposal an i.i.d. sample of T observations drawn from the aforementioned return distribution.

A simple and common approach to portfolio choice treats the statistical objective of estimating return distributions in isolation. First, the investor infers the mean and covariance of returns while disregarding the utility specification. Second, the investor treats estimates $\widehat{\mu}$ and $\widehat{\Sigma}$ as the true return moments and, given these, selects portfolio weights to optimize utility. This is known colloquially as the “plug-in” estimator, because it leads to a solution that replaces inputs to the solution (5.2) with estimated counterparts

$$\widehat{w} = \frac{1}{\gamma}\widehat{\Sigma}^{-1}\widehat{\mu}. \tag{5.3}$$

The motivation of the plug-in solution comes from the fact that, if $\widehat{\mu}$ and $\widehat{\Sigma}$ are consistent estimates, then $\widehat{w}$ is also consistent for w^*. It offers a simple and direct use of machine learning forecasts to achieve the desired economic end. While consistency of $\widehat{w}$ can be an attractive theoretical property, rarely do we have sufficient data for convergence properties to kick-in. When N is large, consistency is in peril. Thus, unfortunately, the plug-in portfolio solution has a tendency to perform poorly in practice, and is disastrous in settings where the number of assets begins to approach the number of training observations (Jobson and Korkie, 1980; Michaud, 1989).

Kan and Zhou (2007) connect the portfolio estimation problem to statistical decision theory by noting that the difference in investor utility arising from the "true" optimal portfolio (w^*) versus the plug-in portfolio ($\widehat{w}$) can be treated as an economic loss function with expected value

$$\mathrm{E}[L(w^*, \widehat{w})|\mu, \Sigma] = U(w^*) - \mathrm{E}[U(\widehat{w})|\mu, \Sigma]. \tag{5.4}$$

This is the well known statistical "risk function," which in the current environment can be viewed as the certainty equivalent cost of using a suboptimal portfolio weight.

Suppose the investor estimates μ and Σ using the sample sufficient statistics

$$\widehat{\mu} = \frac{1}{T}\sum_{t=1}^{T} R_t, \quad \widehat{\Sigma} = \frac{1}{T}\sum_{t=1}^{T}(R_t - \widehat{\mu})(R_t - \widehat{\mu})',$$

then Kan and Zhou (2007) show that the risk function of the plug-in portfolio is

$$\mathrm{E}[L(w^*, \widehat{w})|\mu, \Sigma] = a_1 \frac{\mathrm{SR}^2}{2\gamma} + a_2 \tag{5.5}$$

where SR^2 is the squared Sharpe ratio of the true optimal strategy w^* and (a_1, a_2) are constants that depend only on the number of sample observations T and the number of assets N.[2] The formula shows that the expected loss is zero for w^* and is strictly positive for any other weight. Holding N fixed, both a_1 and a_2 approach zero as T increases, suggesting that the loss is inconsequential when T is large relative to

[2] The explicit formulae are given by $a_1 = 1 - \frac{T}{T-N-2}\left[2 - \frac{T(T-2)}{(T-N-1)(T-N-4)}\right]$, $a_2 = \frac{NT(T-2)}{(T-N-1)(T-N-2)(T-N-4)}$.

N. Holding T fixed, both a_1 and a_2 are increasing in N, meaning that the loss becomes more severe when there are more parameters (more assets), all else equal. And the investor suffers a bigger loss when the Sharpe ratio is higher or risk aversion is lower as both of these lead the investor to tilt toward risky assets, which in turn increases the investor's exposure to estimation uncertainty.

Equation (5.5) makes it possible to quantify the utility cost of the plug-in estimator in reasonable calibrations. These costs suggest large utility shortfalls for investors that unduly rely on consistency properties of $\widehat{w}$. But beyond its high dimensional challenge, a glaring disadvantage of the plug-in solution is its *inadmissibility*. Other portfolio rules, such as those based on shrinkage or Bayesian decision theoretic approaches, improve expected out-of-sample performance for any values of the true moments μ and Σ. The plug-in estimator is in this sense an especially unattractive portfolio solution. Its inadmissibility stems from the failure to account for dependence of end-use utility on parameter estimation.

By placing their analysis in a decision theoretical framework, Kan and Zhou (2007) next provide a straightforward demonstration that investors have tools at their disposal to mitigate out-of-sample utility loss and easily improve over the plug-in portfolio. In particular, an investor can understand ex ante how estimation uncertainty impacts their out-of-sample utility, and how she can internalize this impact with modified portfolio rules that reduce the deleterious effects of estimation noise. Kan and Zhou (2007) demonstrate plug-in portfolio inadmissibility by proving that simple portfolio tweaks (like tilting the plug-in solution toward a heavier weight in the risk-free asset or mixing it with the plug-in minimum variance portfolio) produce higher expected utility for any value of μ and Σ. This insight is the essence of statistical decision theory—outcomes are improved by integrating aspects of the investor's utility objective into the statistical problem of weight estimation, rather than treating estimation and utility maximization as distinct problems.[3]

[3] There is a large literature analyzing portfolio choice through a Bayesian and economic decision theoretical lens (see the survey of Avramov and Zhou (2010)). Early examples such as Jorion (1986) and Frost and Savarino (1986) demonstrate the utility benefits conferred by Bayes-Stein shrinkage solutions. Other contributions in this area use informative and economically motivated priors (Black and Litterman,

But the decision theoretical approach also gives rise to a paradox. The problem formulation above is extraordinary simple. Returns are i.i.d. normal. Preferences are quadratic. There is no aspect of the problem that is ambiguous or unspecified. It would seem, then, that any solution to this problem (decision theoretical or otherwise) must be similarly clear cut and unambiguous. But this is not the case! To see this, let us take stock of what each assumption buys us. The i.i.d. normality assumption implies that sample means and covariances are sufficient statistics for the investor's information set (the sample of T return observations). From this we know that any effective solution to the quadratic utility problem should depend on data through these two statistics and no others, so we know to restrict solutions to the form

$$\widehat{w} = f(\widehat{\mu}, \widehat{\Sigma}). \tag{5.6}$$

Next, quadratic utility plus i.i.d. normality buys us some analytical tractability of the expected loss in (5.5).[4] We seek a portfolio rule that minimizes $\mathrm{E}[L(w^*, \widehat{w})|\mu, \Sigma]$. But, without further guidance on the function f, the problem of minimizing expected loss is ill-posed. Herein lies the paradox. The seemingly complete formulation of the Markowitz problem delivers an unambiguous solution only when the parameters are known. Once estimation uncertainty must be accounted for, the problem provides insufficient guidance for estimating portfolio rules that maximize expected out-of-sample utility.

Some make progress by imposing a specific form of f. For example, to demonstrate inadmissibility of the plug-in solution, Kan and Zhou (2007) restrict f to a set of linear functions and show that within this set there are portfolio solutions that uniformly dominate the plug-in rule. Meanwhile they note that good choices for $\widehat{w}$ *"can potentially be a very complex nonlinear function of $\widehat{\mu}$ and $\widehat{\Sigma}$ and there can be infinitely many ways to construct it. However, it is not an easy matter to determine the*

1992; Pastor, 2000; Pastor and Stambaugh, 2000; Tu and Zhou, 2010). While these are not machine learning methods per se, they paved the way for development of integrated estimation and utility optimization solutions that are now the norm among machine learning portfolio methods.

[4] But this analytical tractability only obtains for certain choices of f (for example, those linear in $\widehat{\mu}$, $\widehat{\Sigma}^{-1}$, or $\widehat{\Sigma}^{-1}\widehat{\mu}$).

optimal $f(\widehat{\mu}, \widehat{\Sigma})$." Tu and Zhou (2010) and Kan *et al.* (2022) elaborate on this idea with alternative portfolio combinations and propose portfolio rules that aim to minimize utility loss under estimation uncertainty. Yuan and Zhou (2022) extend this analysis to the high-dimensional case where $N > T$. Instead of restricting the functional form of portfolio strategies f, Da *et al.* (2022) make progress on this problem by imposing restrictions on the data generating process of returns.

5.2 Integrated Estimation and Optimization

So where does the inadmissibility paradox leave us more generally? We have an unknown function f appearing in an economic problem—this is an ideal opportunity to leverage the strengths of machine learning. In the current setting how might this work? Perhaps we can choose a flexible model like a neural network to parameterize f and search for function parameter values that maximizes expected out-of-sample utility (or, equivalently, minimizes the risk function $\mathrm{E}[L(w^*, \widehat{w})|\mu, \Sigma]$)? The problem is that $\mathrm{E}[L(w^*, \widehat{w})|\mu, \Sigma]$ can only be derived for certain special cases like the plug-in estimator. For general f, the expected out-of-sample utility is not available in closed form, and if it were it depends on the unknown true parameters μ and Σ.

A feasible alternative approach is to choose the portfolio rule f to optimize in-sample utility, and to regularize the estimator to encourage stable out-of-sample portfolio performance (e.g., via cross-validation). This falls neatly into the typical two-step machine learning workflow:

> Step 1. Choose a class of functions indexed by a tuning parameter (for example, the class might be linear ridge models, which are indexed by the ridge penalty z) and use the training sample to estimate model parameters (one set of estimates for each value of z). Take note of how immediately the simple, analytical Markowitz paradigm morphs into a machine learning problem. When the true parameters are known, the tightly specified Markowitz environment gives a simple and intuitive closed-form portfolio solution. Yet with just a sprinkling of estimation uncertainty, the decision theoretical problem is only partially specified and the portfolio rule

that minimizes out-of-sample loss cannot be pinned down. This leads us to use the economic objective as the estimation objective. In other words, the decision theoretical structure of the problem compels us to integrate the statistical and economic objectives and to consider open-minded machine learning specifications.

Step 2. In the validation sample, choose tuning parameters (e.g., a specific ridge penalty z) to optimize expected out-of-sample performance. The theoretical underpinnings of regularization are rooted minimization of estimation risk. In the Kan and Zhou (2007) framework these underpinnings are explicit and regularization is pursued through analytical calculations. The more standard course of action replaces the theoretical calculations with empirics. Expected out-of-sample utility is approximated by realized utility in the validation sample, and the value of z that maximizes validation utility is selected. This empirical cross-validation obviates the need for theoretical simplifications that would be necessary to optimize expected utility directly.

Once the portfolio choice problem is cast in the machine learning frame, one can begin generalizing any otherwise restrictive assumptions. For example, the return distribution need not be normal nor i.i.d., since we do not need to explicitly calculate expected out-of-sample utility. We can thus write the general portfolio rule as

$$\widehat{w} = f(X_T), \tag{5.7}$$

where X_T collates all data relevant to the decision making process. It may include the return sample $\{R_t\}_{t=1}^{T}$ itself as well as any conditioning variables that modulate the conditional return distribution. We can likewise swap out preference specifications to accommodate tail risk concerns, multiple-horizon objectives, or other complications.

5.3 Maximum Sharpe Ratio Regression

Frontier machine learning methods for portfolio choice directly take utility optimization into consideration when estimating portfolio rules.

Important early progress in machine learning portfolio choice comes from Ait-Sahalia and Brandt (2001), Brandt and Santa-Clara (2006), and Brandt *et al.* (2009). Their central contribution is to formalize the portfolio problem as a one-step procedure that integrates utility maximization into the statistical problem of weight function estimation. The approach makes no distributional assumptions for returns. Instead, it specifies investor utility and an explicit functional form for the investor's portfolio weight function in terms of observable covariates. Parameters of the weight function are estimated by maximizing average in-sample utility. Following Brandt (1999), we refer to this as the "parametric portfolio weight" approach.

To keep our presentation concrete, we continue in the spirit of Markowitz and specialize our analysis to the mean-variance utility framework. Our primary motivation for doing so is that we can cast the "parametric portfolio weight" as an OLS regression. To do so, we rely on Theorem 1 of Britten-Jones (1999) who shows that the Markowitz plug-in solution in (5.3) is proportional to the OLS coefficient in a regression of a constant vector on the sample asset returns. In particular, the OLS regression

$$1 = w' R_t + u_t \tag{5.8}$$

delivers coefficient[5]

$$w^{\text{OLS}} \propto \widehat{\Sigma}^{-1}\widehat{\mu}.$$

Intuitively, regression (5.8) seeks a combination of the risky excess returns R_t that behaves as closely as possible to a positive constant. This is tantamount to finding the combination of risky assets with the highest possible in-sample Sharpe ratio. While this is exactly proportional to identifying the in-sample tangency portfolio of risky assets, we will see that the regression formulation is attractive for incorporating machine learning methods into parameterized portfolio problems. Going forward, we dub regressions of the form (5.8) as "maximum Sharpe ratio regression," or MSRR.

Brandt *et al.* (2009) focus on the portfolio weight parameterization

$$w_{i,t} = \bar{w}_{i,t} + s'_{i,t}\beta \tag{5.9}$$

[5] To derive this result, note that $w^{\text{OLS}} = (\widehat{\Sigma} + \widehat{\mu}\widehat{\mu}')^{-1}\widehat{\mu} = \widehat{\Sigma}^{-1}\widehat{\mu}(1 + \widehat{\mu}\widehat{\Sigma}^{-1}\widehat{\mu})^{-1}$.

where $\bar{w}_{i,t}$ is some known benchmark weight of asset i at time t, and $s_{i,t}$ is a set of K signals that enter linearly into the portfolio rule with K-dimensional coefficient vector β. This introduces scope for dynamic portfolio weights that vary with conditioning variables. For notational simplicity, we normalize $\bar{w}_{i,t}$ to zero and stack weights and signals into the vector equation[6]

$$w_t = S_t \beta, \tag{5.10}$$

where w_t is $N \times 1$ and S_t is $N \times K$. Substituting into (5.8), MSRR becomes

$$1 = \beta'(S'_{t-1} R_t) + u_t = \beta' F_t + u_t. \tag{5.11}$$

We may view w_t as a set of dynamic weights on the N base assets R_t. Or, equivalently, β is a set of static weights on K "factors" (characteristic-managed portfolios):

$$F_t = S'_{t-1} R_t. \tag{5.12}$$

In other words, the restriction that the same signal combo applies for all assets immediately converts this into a cross-sectional trading strategy (since F represents portfolios formed as a cross-sectional covariance of returns with each of the signals individually). The estimated portfolio coefficients are $\widehat{\beta} = \widehat{\text{Cov}}(F)^{-1}\widehat{\text{E}}(F)$. In the language of Brandt and Santa-Clara (2006), the MSRR weight parameterization in (5.10) "augments the asset space," from the space of base returns to the space of factors. With this augmentation, we arrive back at the basic problem in (5.8) that estimates fixed portfolio weights on factors. A benefit of the MSRR formulation is that we have simple OLS test statistics at our disposal for evaluating the contribution of each signal (as originally proposed by Britten-Jones, 1999).

Brandt (1999) summarizes the benefits of parametric portfolio rules, noting

> *"the return modeling step is without doubt the Achilles' heel of the traditional econometric approach.... Focusing directly*

[6] Brandt *et al.* (2009) also note the ease of accommodating industry neutrality or other weight centering choices as well as handling time-varying cross-section sizes (in which case they recommend the normalization $\frac{1}{N_t} s'_{i,t}\beta$ in (5.9), where N_t is the number of assets at period t). We abstract from these considerations for notational simplicity.

> *on the optimal portfolio weights therefore reduces considerably the room for model mispecification and estimation error."*

They emphasize that estimation error control stems from a highly culled parameter dimension (relative to estimating all assets' means and covariances).

The Brandt *et al.* (2009) empirical application studies portfolio rules for thousands of stocks using only three parameters (that is, conditioning only on stocks' size, value, and short-term reversal characteristics). The least squares formulation for parameterized portfolio rules prompts a brainstorm of adaptations to incorporate machine learning structures. We are immediately tempted to consider large K regressions for incorporating a rich set of conditioning information, including for example the full zoo of factor signals and perhaps their interactions with macroeconomic state variables, along with the usual machine learning regularization strategies for shrinkage and model selection. The idea of lasso-regularized parametric portfolios is broached by Brandt and Santa-Clara (2006) and Brandt *et al.* (2009), and thoroughly investigated by DeMiguel *et al.* (2020).[7] The beauty of MSRR is that efficient software for lasso and elastic-net regression can be deployed for the portfolio choice problem straight off-the-shelf. This covers essentially all varieties of penalized regression that employ ℓ_1 or ℓ_2 parameter penalties, which includes a range of realistic portfolio considerations like leverage control (often formulated as an ℓ_1 penalty) or trading costs adjustments (often modeled as an ℓ_2 penalty). In the special case of ridge regression with penalty z, the penalized MSRR objective is

$$\min_{w} \sum_{t} (1 - \beta' F_t)^2 + z\beta'\beta \tag{5.13}$$

with solution

$$\widehat{\beta} \propto (\widehat{\text{Cov}}(F) + zI)^{-1} \widehat{\text{E}}(F), \tag{5.14}$$

which modifies the standard sample tangency portfolio of factors by shrinking the sample covariance toward identity. This naturally connects

[7] Ao *et al.* (2018) also study a related lasso-regularized portfolio choice problem that they call MAXSER.

with the broader topic of covariance shrinkage for portfolio optimization discussed by Ledoit and Wolf (2004) and Ledoit and Wolf (2012).

Together with the ease of manipulating MSRR specifications, the viability of penalized linear formulations raises exciting prospects for parametric portfolio design. For example, we can break the restriction of identical coefficients for all assets in (5.10) with the modification

$$w_{i,t} = s'_{i,t}\beta_i \tag{5.15}$$

and associated regression representation

$$1 = \text{vec}(B)'\{\text{vec}(S_{t-1}) \odot (R_t \otimes 1_K)\} + u_t, \tag{5.16}$$

where $B = [\beta_1, ..., \beta_N]$. This regression has the interpretation of finding the tangency portfolio among NK "managed" strategies, with each individual strategy an interaction of return on asset i with one of its signals. A minor modification of this problem makes it possible to accommodate assets with different sets of predictors. As a twist on the applicability of penalized MSRR, one interesting approach would be to penalize $\beta_i - \bar{\beta}$, which allows some heterogeneity across assets but shrinks all assets' weight rules toward the average rule, in the spirit of empirical Bayes.

Brandt *et al.* (2009) discusses a number of extensions and refinements to their parametric portfolio problem, including non-negative transformations of weights for long-only strategies, trading cost considerations, and multi-period investment horizons. Most of these extensions can be folded into the MSRR framework outlined here as long as the mean-variance objective is retained. For more general utility functions, estimation must step out of the OLS regression framework and requires numerical optimization. This is more computationally challenging for high-dimensional parameterization but adds very little conceptual complexity.

5.4 High Complexity MSRR

MSRR is adaptable to sophisticated machine learning models such as neural networks. One simple example of a potential MSRR network architecture is shown in Figure 5.1. A set of L "raw" signals for stock

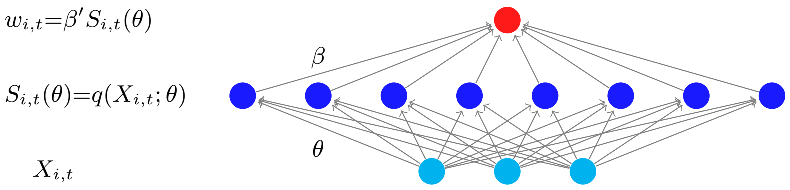

$w_{i,t}{=}\beta' S_{i,t}(\theta)$
β
$S_{i,t}(\theta){=}q(X_{i,t};\theta)$
θ
$X_{i,t}$

Figure 5.1: MSRR Neural Network

Note: Illustration of a simple neural architecture mapping conditioning variables $X_{i,t}$ into optimal portfolio weights $w_{i,t}$.

i, denoted $X_{i,t}$, are inputs to the network. These propagate through a sequence of nonlinear transformations (parameterized by network coefficients θ) to produce a vector of K output signals, $S_{i,t}(\theta) = q(X_{i,t};\theta)$ where $q \in \mathbb{R}^L \to \mathbb{R}^K$. Then an otherwise standard MSRR formulation uses the transformed characteristics to derive the mean-variance efficient portfolio rule:

$$1 = \beta' S_{t-1}(\theta)' R_t + u_t. \tag{5.17}$$

Estimation of the parameters (β, θ) is a nonlinear least squares problem that is straightforward to implement with off-the-shelf neural network software packages. The interpretation of this estimation is that it searches for flexible nonlinear transformations of the original signals set to engineer new asset characteristics that can actively manage positions in the base assets and produce high Sharpe ratio strategies.

Simon *et al.* (2022) pursue this framework using a feed-forward neural network in the portfolio weight function with three hidden layers of 32, 16, and 8 neurons in each layer, respectively. They show that the neural network enhances parametric portfolio performance. The linear portfolio specification with the same predictors earns an annualized out-of-sample Sharpe ratio of 1.8, while the neural network portfolio achieves a Sharpe ratio of 2.5, or a 40% improvement.

Didisheim *et al.* (2023) prove theoretically that the “virtue of complexity” also holds in mean-variance efficient portfolio construction. Under weak assumptions, the Sharpe ratio of a machine learning portfolio is increasing in model parameterization. Empirically, they present one especially convenient neural specification derived within the framework of Section 2. In particular, they propose a neural architecture using random feature ridge regression. The formulation is identical to (5.17) except that the neural coefficients θ are randomly generated rather than estimated. As predicted by theory, the realized out-of-sample Sharpe ratio in their empirical analysis is increasing with the number of model parameters and flattens out around 4 when 30,000 parameters are used.

5.5 SDF Estimation and Portfolio Choice

The equivalence between portfolio efficiency and other asset pricing restrictions, like zero alphas in a beta pricing model or satisfaction of SDF-based Euler equations, imply that there are statistical objectives (besides utility objectives) to work from when estimating optimal portfolios. Much of the financial machine learning literature focuses on SDF estimation. Knowledge of the SDF is interesting for a number of economic questions, such as inferring investor preferences, quantifying pricing errors, and eliciting the key sources of risk that influence asset prices. In practice, however, the financial machine learning literature has placed insufficient restrictions on the SDF estimation problem to answer questions about specific economic mechanisms. Instead, this literature tends to evaluate estimation results in terms of the out-of-sample Sharpe ratio of the estimated SDF. That is, machine learning approaches to SDF estimation focus on the mean-variance portfolio choice problem, motivated by Hansen and Richard (1987)'s theorem of SDF and mean-variance frontier equivalence.

The typical formulation of the SDF estimation problem is the following. If an SDF exists (given, for example, the absence of arbitrage), it can be represented as a portfolio w of excess returns (barring a constant term)

$$M_t = 1 - w' R_t. \tag{5.18}$$

Furthermore, an SDF must satisfy the standard investor Euler equation at the heart of asset pricing theory:

$$\mathrm{E}[M_t R_t] = 0. \tag{5.19}$$

Combining these we see:

$$\mathrm{E}[R_t - R_t R_t' w] = 0. \tag{5.20}$$

This equation identifies a tradable SDF. Note that it is also the first-order condition to the Britten-Jones (1999) MSRR problem,

$$\min_{w} \mathrm{E}\left[(1 - w'R_t)^2\right],$$

which formally connects the problems of SDF estimation and Sharpe ratio maximization, and thus it results in the tangency portfolio as the estimated SDF weights.

Kozak *et al.* (2020) propose an SDF estimation problem in a conditional and parameterized form that has a close connection to MSRR. In particular, they posit an SDF portfolio $w_t = S_t\beta$ just as in (5.10), and note that this implies an SDF of the form

$$M_t = 1 - \beta' F_t \tag{5.21}$$

with F_t defined in (5.12).[8] In other words, the SDF that prices assets conditionally can be viewed as a static portfolio of characteristic-managed factors. The main contribution of Kozak *et al.* (2020) is to introduce regularization into the conditional SDF estimation problem. They primarily analyze ridge regularization, which directly maps to the MSRR ridge estimator in equation (5.14). Rotating their estimator into the space of principal components of the factors, Kozak *et al.* (2020) show that ridge shrinkage induces heavier shrinkage for lower ranked components. They offer an insightful take on this derivation: "*The economic interpretation is that we judge as implausible that a PC with low eigenvalue could contribute substantially to the volatility of the SDF and hence to the overall maximum squared Sharpe ratio.*" Kozak (2020) uses a kernel method to estimate a high dimensional SDF with a clever implementation of the "kernel trick" that circumvents the need to estimate an exorbitant number of parameters.

The empirical results of Kozak *et al.* (2020) indicate that their methodology of estimating a factor-based SDF with penalized regression delivers a portfolio with economically potent (and statistically significant) out-of-sample performance relative to standard benchmark

[8] Kozak *et al.* (2020) normalize the SDF to have mean 1: $M_t = 1 - \beta'(F_t - \mathrm{E}(F_t))$. We adopt a different normalization to conform to the MSRR problem in Britten-Jones (1999).

asset pricing models. Constructing the SDF from 50 anomaly factor portfolios results in an SDF with an annualized information ratio of 0.65 versus the CAPM. If instead of using 50 characteristics for their set of signals, they supplement the raw characteristics with second and third powers of those characteristics as well as pairwise characteristic interactions (for a total of 1,375 signals), this information ratio increases to 1.32.[9]

Giglio *et al.* (2021b) analyze and compare the asymptotic properties of Kozak *et al.* (2020)'s estimator with PCA and RP-PCA-based estimators proposed by Giglio and Xiu (2021) and Lettau and Pelger (2020a), in a common unconditional factor model framework (4.1). With $\widehat{\gamma}$ and $\widehat{V}$ defined by (4.6) and (4.4) the PCA-based SDF estimator is given by

$$\widehat{M_t} = 1 - \widehat{\gamma}'\widehat{V}_t.$$

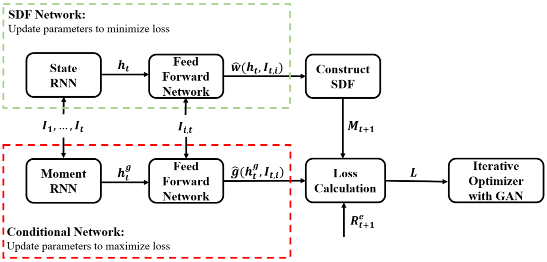
Giglio *et al.* (2021b) show that the ridge and PCA-based estimators are consistent as long as factors are pervasive.

Unlike Kozak *et al.* (2020) who model the SDF as a portfolio of anomaly factors, Chen *et al.* (2021) extend the SDF estimation problem by modeling weights on individual stocks. Their model deviates from MSRR and the parametric portfolio problem in a few interesting ways. First, they allowing for a flexible weight function that includes a recurrent neural network component. Second, they formulate their estimator via GMM with a sophisticated instrumentation scheme. Beginning from a conditional version of the basic Euler equation in (5.20), Chen *et al.* (2021) rewrite the GMM objective function as

$$\min_{w} \max_{g} \frac{1}{N} \sum_{j=1}^{N} \left\| \mathrm{E}\left[\left(1 - \sum_{i=1}^{N} w(S_{M,t}, S_{i,t}) R_{i,t+1} \right) R_{j,t+1} g(S_{M,t}, S_{j,t}) \right] \right\|. \tag{5.22}$$

The function $w(S_{M,t}, S_{i,t})$ describes the weight of the i^{th} stocks in the tradable SDF, which is a scalar function of macroeconomic predictors ($S_{M,t}$, which influence all assets' weights) and stock-specific predictors ($S_{i,t}$, which only directly influence the weight of stock i). It uses an LSTM to capture dynamic behavior of $S_{M,t}$, whose output flows into a

[9] Information ratios are inferred from alphas and standard errors reported in Table 4 of Kozak *et al.* (2020).



This figures shows the model architecture of GAN (Generative Adversarial Network) with RNN (Recurrent Neural Network) with LSTM cells. The SDF network has two parts: (1) A LSTM estimates a small number of macroeconomic states. (2) These states together with the firm-characteristics are used in a FFN to construct a candidate SDF for a given set of test assets. The conditioning network also has two networks: (1) It creates its own set of macroeconomic states, (2) which it combines with the firm-characteristics in a FFN to find mispriced test assets for a given SDF M. These two networks compete until convergence, that is neither the SDF nor the test assets can be improved.

Figure 5.2: Chen *et al.* (2021)'s Network Architecture

Note: Source: Chen *et al.* (2021).

feed-forward network where it is combined with the stock-level predictors to produce the final portfolio weight output (see the top portion of Figure 5.2).

The function $g(S_{M,t}, S_{j,t})$ is a set of D instrumental variables which provide the system of moment conditions necessary to estimate the portfolio weight function. The instrumental variables, however, are derived from the underlying data via the network function, g. The structure of g mirrors the structure of w. Macroeconomic dynamics are captured with an LSTM, and the output is combined with stock-specific variables $S_{j,t}$ in a feed-forward network to produce instruments for the Euler equation pricing errors of asset j. The loss function is the norm of the pricing errors aggregated across all stocks, which is minimized to arrive at weight function estimates.

Perhaps the most interesting aspect of this specification is that instrumental variables are generated in an adversarial manner. While the estimator searches for an SDF weight function that minimizes pricing errors for given a set of instruments (the outer minimization objective in (5.22)), it simultaneously searches for the instrumental variables function that casts the given SDF in the worst possible light (the inner minimization objective in (5.22)).

Motivated by these empirical studies of machine learning SDFs, Didisheim *et al.* (2023) theoretically analyze the role that model complexity plays shaping the properties of SDF estimators. Like Kelly *et al.* (2022a), they focus on high-dimensional ridge estimators with two main differences. First, they move from a single asset time series setting with a single asset to a panel setting with an arbitrary number of risky assets. Second, they reorient the statistical objective from time series forecasting to SDF optimization. In this setting, Didisheim *et al.* (2023) explicitly derive an SDF's expected out-of-sample Sharpe ratio and cross-sectional pricing errors as a function of its complexity. Their central result is that expected out-of-sample SDF performance is strictly improving in SDF model complexity when appropriate shrinkage is employed. They report empirical analyses that align closely with their theoretical predictions. Specifically, they find that the best empirical asset pricing models have an extremely large number of factors (more than the number of training observations or base assets).

5.5.1 Non-tradable SDF Estimation

The preceding section focuses on estimating an SDF represented in the space of excess returns, or as a tradable portfolio. As noted, the rather minimal economic structure imposed in these estimation problems make it difficult to investigate economic mechanisms. A smaller but equally interesting literature infers an SDF by balancing structure and flexibility, including using (semi-)nonparametric model components embedded in a partial structural framework and conducting hypothesis tests.

It is fascinating to recognize that the idea of parameterizing asset pricing models using neural networks appears as early in the literature as Bansal and Viswanathan (1993). These prescient specifications have all of the basic ingredients that appear in more recent work, but are constrained by smaller data sets and thus study small network specifications.

Chen and Ludvigson (2009) is perhaps the finest example to date of this approach. Their model environment is broadly rooted in the nonlinear habit consumption utility specification of Campbell and Cochrane (1999) and related models of Menzly *et al.* (2004), Wachter (2006), and Yogo (2006). As the authors rightly point out, the variety of habit specifications in the literature suggest "*the functional form of the habit should be treated, not as a given, but as part and parcel of any empirical investigation,*" and in turn they pursue a habit model by "*placing as few restrictions as possible on the specification of the habit and no parametric restrictions on the law of motion for consumption.*"

The model of Chen and Ludvigson (2009) is best summarized by its specification of habit consumption, which is the key driver of investors' marginal utility. In particular, habit consumption is

$$X_t = C_t g\left(\frac{C_{t-1}}{C_t}, \ldots, \frac{C_{t-L}}{C_t}\right),$$

where C_t is consumption level at time t and g is the "habit function" that modulates the habit level as a function of consumption in the recent L periods. Habit impacts investor well-being via the utility function

$$U = \mathrm{E}\left(\sum_{t=0}^{\infty} \delta^t \frac{(C_t - X_t)^{1-\gamma} - 1}{1-\gamma}\right),$$

which in turn defines the SDF (where δ and γ are the time discount factor and risk aversion, respectively). In contrast to Campbell and Cochrane (1999) who specify an explicit functional form for g, Chen and Ludvigson (2009) use a feed-forward neural network as a generic approximating model for the habit function.

Like Chen *et al.* (2021), they estimate an SDF by minimizing the norm of model-implied Euler equation conditional moments with macro-finance data as instruments. A particularly appealing aspect of the analysis is the care taken to choose the size of the neural network and conduct statistical tests in line with asymptotic theory upon which their tests rely. First, they extend the asymptotic theory of Ai and Chen (2003) and Ai and Chen (2007) for "sieve minimum distance (SMD)" estimation of neural networks to accommodate the serial correlation in asset pricing data. Ultimately, they decide on a small model by machine learning standards (only 18 parameters), but with a sample of only 200 observations this restraint lends credibility to their tests.

The payoff from careful asymptotic analysis is the ability to conduct hypothesis tests for various aspects of habit persistence in investor preferences. Results favor a nonlinear habit function over linear, and internal habit formation (i.e., dictated by one's own consumption history) rather than external habit (or "keeping up with the Joneses"). Lastly, they conduct a comparison of pricing errors from their estimated habit model versus other leading asset pricing models such as the three-factor model (Fama and French, 1993) and the log consumption-wealth ratio CAPM model (Lettau and Ludvigson, 2001), and find significant improvements in pricing of size and book-to-market sorted portfolios with neural network habit formulation.

5.5.2 Hansen-Jagannathan Distance

The GMM (and the related SMD) SDF estimation approach is rooted in the problem of minimizing squared pricing errors. The pricing error is defined by the Euler equation, and may be represented as the average discounted excess return (which should be zero according to the Euler equation) or as the difference between the undiscounted average excess return and the model-based prediction of this quantity. While it is convenient to directly compare models in terms of their pricing errors, Hansen and Jagannathan (1997) emphasize that the efficient GMM estimator uses a model-dependent weighting matrix to aggregate pricing errors, which means we cannot compare models in terms of their optimized GMM loss function values. As a solution to the model comparison problem, Hansen and Jagannathan (1997) recommend evaluating all models in terms of a common loss function whose weighting matrix is independent of the models. In particular, they propose a distance metric of the form

$$HJ_m^2 = \min_{\theta_m} e(\theta_m)'\tilde{\Sigma}^{-1}e(\theta_m), \tag{5.23}$$

where $e(\theta_m)$ is the vector of pricing errors associated with the m^{th} model (which is parameterized by θ_m). The square root of equation (5.23) is known as the "HJ-distance." There are two key aspects to note in its formulation. First, pricing errors are weighted by $\tilde{\Sigma} = \frac{1}{T}\sum_t R_t R_t'$, which is model independent, and thus puts all models on equal footing for comparison. Second, the pricing errors are based on the parameters that minimize the HJ-distance, as opposed to those dictated by the efficient GMM objective.

While these differences versus the standard GMM objective function are subtle, they equip the HJ-distance with some extraordinary theoretical properties. First, Hansen and Jagannathan (1997) show that HJ_m is equal to the pricing error of the *most mispriced* portfolio of base assets (those corresponding to the vector e) arising from model m. That is, the HJ-distance describes the best that model m can do in minimizing the worst-case pricing error.[10] Second, the HJ-distance describes the least squares distance between the (potentially misspecified) SDF of model m and the SDF family that correctly prices all assets.

Both of these properties imply that the HJ-distance is a particularly attractive objective function for training machine learning models of the SDF and optimal portfolios. A machine learning model that minimizes the HJ-distance will be robust (as it seeks to minimize worst-case model performance) and will deliver a model that is minimally misspecified in an ℓ_2 sense. Training machine learning models via HJ-distance aligns with Ludvigson (2013)'s admonition

> *"for greater emphasis in empirical work on methodologies that facilitate the comparison of competing misspecified models, while reducing emphasis on individual hypothesis tests of whether a single model is specified without error."*

Progress has begun in integrating the HJ-distance into financial machine learning problems. Kozak *et al.* (2020) link their SDF estimation approach to an HJ-distance minimization problem with a ridge (or lasso) penalty. Theirs can be viewed as an unconditional HJ-distance applied to the space of factors. Recently, Gagliardini and Ronchetti (2019), Antoine *et al.* (2018), and Nagel and Singleton (2011) analyze approaches to model comparison via a *conditional* HJ-distance. Didisheim *et al.* (2023) derive the theoretical behavior of the HJ-distance in complex machine learning models and show that the distance is decreasing in model complexity. There remains large scope for future research into machine learning models trained through the conditional or unconditional HJ-distance.

[10] This worst-case portfolio is essentially the "Markowitz portfolio" of pricing errors, achieving the highest pricing error per unit of volatility.

5.6 Trading Costs and Reinforcement Learning

The financial machine learning literature provides a flexible framework to combine several characteristics into a single measure of overall expected returns (e.g. Gu *et al.*, 2020b). The same literature documents the relative "feature importance" of different return prediction characteristics (e.g. Chen *et al.*, 2021). These findings suggest that the prediction success of machine learning methods is often driven by short-lived characteristics that work well for small and illiquid stocks e.g. Avramov *et al.*, 2022a, suggesting that they might be less important for the real economy (e.g. Van Binsbergen and Opp, 2019). The high transaction costs of portfolio strategies based on machine learning 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 machine learning expected return estimates merely tell us about mispricings that investors do not bother to arbitrage away because the costs are too large, the mispricing too fleeting, and the mispriced markets too small to matter? Or, do trading-cost-aware machine learning predictions also work for large stocks, over significant time periods, and in a way that matters for many investor types, thus leading to new and important economic insights?

It is common in financial research to separate the question of portfolio implementability from the return prediction (or portfolio weight estimation) component of the problem. In this approach, the estimation problem abstracts from transaction costs and turnover, and it is not uncommon for the resultant investment strategies to produce negative returns net of transaction costs. There is a natural explanation for this result. It's not that the market doesn't know predictive patterns are there; it's that the patterns are there *because* they are too costly to trade or due to other limits-to-arbitrage. This is a critical challenge for predictive methods in finance. Without somehow embedding limits-to-arbitrage into the statistical prediction model, the model will tend to isolate predictive patterns that are unrealistic to trade because these are the ones currently unexploited in the market. This does not rule out the possibility that there are predictable phenomena that may be practically exploited; i.e., predictability that is profitable net of trading costs. It just means that without guidance on costs, a statistical model cannot distinguish between implementable and non-implementable predictive patterns. So, if it's the case that the most prominent predictable patterns in returns are those that are associated with limits-to-arbitrage (a plausible scenario given the unsparing competition in securities markets), these will be the first ones isolated by ML prediction models.

This section discusses attempts to derive machine learning portfolios that can be realistically implemented by market participants with large assets under management, 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 likely to be informative about equilibrium discount rates.

5.6.1 Trading Costs

The literature on portfolio choice in the face of trading costs typically solves the problem by assuming that the trading cost function is known or that trading cost data is available. This obviates the need for an "exploration" step in a machine learning portfolio choice algorithm, keeping the problem more or less in a standard supervised learning framework and avoiding reinforcement learning machinery. Brandt *et al.* (2009) take this approach in the baseline parameterized portfolio setup. Recently, Jensen *et al.* (2022) introduce a known trading cost function into the portfolio choice objective, then use a combination of machine learning and economic restrictions to learn the optimal portfolio rule. We outline their model and learning solution here.

The finance literature has long wrestled with the technical challenges of trading cost frictions in portfolio choice (e.g. Balduzzi and Lynch, 1999; Lynch and Balduzzi, 2000). Garleanu and Pedersen (2013) derive analytical portfolios amid trading costs. Their key theoretical result is that, in the presence of trading costs, optimal portfolios depend not just on one period expected returns $\mathrm{E}_t[R_{t+1}]$, but on expectations of returns over all future horizons, $\mathrm{E}_t[R_{t+\tau}]$ $\tau = 1, 2, ...$[11] Because of this,

[11] Intuitively, the investor prefers to trade on long-lived predictive patterns because the cost of trading can be amortized over many periods, while short-lived predictability

the idea of approaching the return prediction problem with machine learning while accounting for trading costs faces tension. Garleanu and Pedersen (2013) achieve a tractable portfolio solution by linking all horizons together with a linear autoregression specification, which keeps the parameterization small and collapses the sequence of expectations collapses to a linear function of prevailing signal values. Machine learning typically seeks flexibility in the return prediction model; but if separate return prediction models are needed for all future horizons the environment quickly explodes into an intractable specification even by machine learning standards. How can we maintain flexibility in the model specification without making restrictive assumptions about the dynamics of expected returns?[12]

Jensen *et al.* (2022) resolve this tension with three key innovations. First, they relax the linearity of expected returns in predictive signals and allow for nonlinear prediction functions. Second, they relax the autoregressive expected return assumption of Garleanu and Pedersen (2013) by requiring stationary but otherwise unrestricted time series dynamics. From these two solutions, they are able to theoretically derive a portfolio rule as a generic function of conditioning signals. Their third contribution is to show that a component of this portfolio rule (the "aim" function to which the investor gradually migrates her portfolio) can be parameterized with a neural network specification, which sidesteps the aforementioned difficulty of requiring different machine learning return prediction models for different investment horizons.[13] Their solution is an example of a model that combines the benefits of economic structure with the flexibility of machine learning. The economics appear in the form of a solution to the investor's utility maximization problem, which imposes a form of dynamic consistency on the portfolio rule. Only after this step is the flexible machine learning function injected into the portfolio rule while maintaining the economic coherence of the rule.

is quickly cannibalized due to frequent portfolio adjustments.

[12] Another complementary tension in this setting is that, if a trading cost portfolio solution must build forecasts for many return horizons, the number of training observations falls as the forecast horizon increases, so there is generally little data to inform long horizons forecasts.

[13] This requires the assumption that signals are Markovian, but their dynamics need not be further specified.

Specifically, the investor seeks a portfolio w_t to maximize net-of-cost quadratic utility:

$$\text{utility} = \mathrm{E}\left[\mu(s_t)'w_t - \frac{\kappa_t}{2}\left(w_t - g_t w_{t-1}\right)'\Lambda\left(w_t - g_t w_{t-1}\right) - \frac{\gamma_t}{2} w_t'\Sigma w_t\right] \tag{5.24}$$

where $\mu(s_t)$, Σ and Λ summarize asset means, covariances, and trading costs, respectively. These vary over time in predictable ways related to the conditioning variable set s_t, though for purposes here we discuss the simplified model with static covariances and trading costs. κ_t is investor assets under management (the scale of the investor's portfolio is a first-order consideration for trading costs) and g_t is wealth growth from the previous period. Proposition 3 of Jensen *et al.* (2022) shows that the optimal portfolio rule in the presence of trading costs is

$$w_t \approx m g_t w_{t-1} + (I - m)A_t \tag{5.25}$$

where the investor partially trades from the inherited portfolio w_{t-1} toward the "aim" portfolio A_t at time t:

$$A_t \;=\; (I - m)^{-1}\sum_{\tau=0}^{\infty}(m\Lambda^{-1}\bar{g}\Lambda)^{\tau}\, c\mathrm{E}_t\Bigg[\underbrace{\frac{1}{\gamma}\Sigma^{-1}\mu(s_{t+\tau})}_{\text{Markowitz}_{t+\tau}}\Bigg]. \tag{5.26}$$

The aim portfolio depends on m, a nonlinear matrix function of trading costs and covariances, as well as the mean portfolio growth rate $\bar{g} = \mathrm{E}[g_t]$. The aim portfolio is an exponentially smoothed average of of period-by-period Markowitz portfolios, which reflects the fact that adjustments are costly so it is advantageous to smooth portfolio views over time. The portfolio solution may be rewritten as an infinite sum of past aim portfolios and their growth over time:

$$w_t \;=\; \sum_{\theta=0}^{\infty}\left(\prod_{\tau=1}^{\theta} m\, g_{t-\tau+1}\right)(I - m)A(s_{t-\theta})\,. \tag{5.27}$$

This solution in (5.27) embeds all of the relevant dynamic economic structure necessary for an optimal portfolio in the presence of trading costs. But it also shows that, if we substitute this solution into the original utility maximization problem, the learning task may be reduced to finding the aim portfolio function $A(\cdot)$ that maximizes expected utility. Based on this insight, Jensen *et al.* (2022) use the random Fourier features neural network specification (as in Kelly *et al.*, 2022a; Didisheim *et al.*, 2023), but rather than approximating the return prediction function they directly approximate the aim function, $A(\cdot) = f(s_t)\beta$ for f a set of P known random feature functions and β is the regression coefficient vector in $\mathbb{R}^P$. Empirically, their one-step, trading-cost-aware portfolio learning algorithm delivers superior net-of-cost performance compared to standard two-step approaches in the literature that first learn portfolios agnostic of trading costs then smooth the learned portfolio over time to mitigate costs.

5.6.2 Reinforcement Learning

Reinforcement learning broadly consists of machine learning models for solving a sequential decision making problem under uncertainty. They are especially powerful for modeling environments in which an agent takes an action to maximize some cumulative reward function but the state of the system and thus the distribution of future outcomes is influenced by the agent's action. In this case, the agent's choices are key conditioning variables for learning the payoff function. But since most state/action pairs are not observed in the past, the data can be thought of as largely unlabeled, and thus the model does not lend itself to direct supervision. Because relevant labels are only precipitated by the agent's actions, much of reinforcement learning splits effort between learning by experimentation ("exploration") and optimizing future expected rewards given what they've learned ("exploitation").

That the agent seeks to optimize future rewards means that reinforcement learning may be a valuable tool for portfolio choice. However, in the basic portfolio choice problem outlined earlier, the state of the system is not influenced by the investor's decisions. The investor is treated as a price taker, so any portfolio she builds does not perturb the dynamics of risk and return going forward. Thus, given data on realized returns of the base assets, machine learning portfolio choice can be pursued with supervised learning techniques. In other words, the benefits of reinforcement learning for a price taker may be limited.

Of course, the price-taker assumption is unrealistic for many investors. Financial intermediaries such as asset managers and banks are often the most active participants in financial markets. Their prominence as "marginal traders" and their tendency to trade large volumes means that their portfolio decisions have price impact and thus alter the state of the system. Such an investor must learn how their decisions affect market dynamics, and this introduces an incentive to experiment with actions as part of the learning algorithm. That is, once investors must internalize their own price impact, the tools of reinforcement learning are well-suited to the portfolio choice problem.

The mainstream finance literature has seen minimal work on reinforcement learning for portfolio choice,[14] though there is a sizable computer science literature on the topic. The more prominent is an investor's price impact in dictating their future rewards, the more valuable reinforcement learning methods become. As such, the finance profession's tendency to focus on relatively low frequency investment strategies makes it somewhat unsurprising that reinforcement learning has not taken hold, though we expect this to change in coming years. Meanwhile, the computer science literature has largely applied reinforcement learning to higher frequency portfolio problems related to market making and trade execution. While we do not cover this literature here, for interested readers we recommend the survey of Hambly *et al.* (2022).

[14] Cong *et al.* (2020) is one example.

6

Conclusions

We attempt to cover the burgeoning literature on financial machine learning. A primary goal of our work is to help readers recognize machine learning as an indispensable tool for developing our understanding of financial markets phenomena. We emphasize the areas that have received the most research attention to date, including return prediction, factor models of risk and return, stochastic discount factors, and portfolio choice.

Unfortunately, the scope of this survey has forced us to limit or omit coverage of some important financial machine learning topics. One such omitted topic that benefits from machine learning methods is risk modeling. This includes models of conditional variances and covariances, and in particular the modeling of high-dimensional covariance matrices. A large literature uses machine learning methods to improve estimation of covariance matrices and, in turn, improve the performance of optimized portfolios. Closely related to risk modeling is the topic of derivatives pricing. In fact, some of the earliest applications of neural networks in finance relate to options pricing and implied volatility surfaces. Prominent examples of machine learning or nonparametric models for derivatives research include Ait-Sahalia and Lo (1998) and

Ait-Sahalia and Lo (2000), Anders *et al.* (1998), Rosenberg and Engle (2002), Bollerslev and Todorov (2011), and Israelov and Kelly (2017), among many others.

Most research efforts to date involve machine learning to improve performance in prediction tasks. This is the tip of the iceberg. One critical direction for next generation financial machine learning analysis is to better shed light on economic mechanisms and equilibria. Another is using machine learning methods to solve sophisticated and highly nonlinear structural models. Separately, the evolutionary nature of economies and markets, shaped by technological change and shifting regulatory environments, presents economists with the challenge of modeling structural change. An exciting potential direction for research is to leverage the flexible model approximations afforded by machine learning to better detect and adapt to structural shifts.

While this survey has focused on the asset pricing side of finance, machine learning is making inroads in other fields such as corporate finance, entrepreneurship, household finance, and real estate. For example, Hu and Ma (2020) use machine learning techniques to process video pitches of aspiring entrepreneurs to test the role of speaker persuasiveness in early stage financing success. Li *et al.* (2021) use textual analysis of earnings calls to quantify corporate culture and its impact on firm outcomes. Lyonnet and Stern (2022) use machine learning to study how venture capitalists make investment decisions. Erel *et al.* (2021) provide evidence that machine learning algorithms can help firms avoid bad corporate director selections. Fuster *et al.* (2022) quantify the equilibrium impact of machine learning algorithms in mortgage markets. While machine learning is has seen far more extensive use in asset pricing, its further application to corporate finance problems is an exciting area of future research.

References

Ahn, S. C. and J. Bae. (2022). “Forecasting with Partial Least Squares When a Large Number of Predictors are Available”. *Tech. rep.* Arizona State University and University of Glasgow.

Ai, C. and X. Chen. (2003). “Efficient Estimation of Models with Conditional Moment Restrictions Containing Unknown Functions”. *Econometrica.* 71(6): 1795–1843.

Ai, C. and X. Chen. (2007). “Estimation of possibly misspecified semiparametric conditional moment restriction models with different conditioning variables”. *Journal of Econometrics.* 141(1): 5–43.

Ait-Sahalia, Y. and M. W. Brandt. (2001). “Variable Selection for Portfolio Choice”. *The Journal of Finance.* 56: 1297–1351.

Ait-Sahalia, Y., J. Fan, L. Xue, and Y. Zhou. (2022). “How and when are high-frequency stock returns predictable?” *Tech. rep.* Princeton University.

Ait-Sahalia, Y., J. Jacod, and D. Xiu. (2021). “Continuous-Time Fama-MacBeth Regressions”. *Tech. rep.* Princeton University and the University of Chicago.

Ait-Sahalia, Y., I. Kalnina, and D. Xiu. (2020). “High Frequency Factor Models and Regressions”. *Journal of Econometrics.* 216: 86–105.

Ait-Sahalia, Y. and A. W. Lo. (1998). “Nonparametric estimation of state-price densities implicit in financial asset prices”. *The journal of finance.* 53(2): 499–547.

Ait-Sahalia, Y. and A. W. Lo. (2000). “Nonparametric risk management and implied risk aversion”. *Journal of econometrics*. 94(1-2): 9–51.

Ait-Sahalia, Y. and D. Xiu. (2017). “Using Principal Component Analysis to Estimate a High Dimensional Factor Model with High-Frequency Data”. *Journal of Econometrics*. 201: 388–399.

Ait-Sahalia, Y. and D. Xiu. (2019). “Principal Component Analysis of High Frequency Data”. *Journal of the American Statistical Association*. 114: 287–303.

Allen-Zhu, Z., Y. Li, and Z. Song. (2019). “A convergence theory for deep learning via over-parameterization”. In: *International Conference on Machine Learning*. PMLR. 242–252.

Altman, E. I. (1968). “Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy”. *The Journal of Finance*. 23(4): 589–609.

Anders, U., O. Korn, and C. Schmitt. (1998). “Improving the pricing of options: A neural network approach”. *Journal of forecasting*. 17(5-6): 369–388.

Andersen, T. G. and T. Bollerslev. (1998). “Answering the Skeptics: Yes, Standard Volatility Models do Provide Accurate Forecasts”. *International Economic Review*. 39: 885–905.

Andersen, T. G., T. Bollerslev, F. X. Diebold, and P. Labys. (2001). “The Distribution of Exchange Rate Realized Volatility”. *Journal of the American Statistical Association*. 96: 42–55.

Antoine, B., K. Proulx, and E. Renault. (2018). “Pseudo-True SDFs in Conditional Asset Pricing Models*”. *Journal of Financial Econometrics*. 18(4): 656–714. ISSN: 1479-8409. DOI: 10.1093/jjfinec/nby017. eprint: https://academic.oup.com/jfec/article-pdf/18/4/656/35053168/nby017.pdf. URL: https://doi.org/10.1093/jjfinec/nby017.

Ao, M., L. Yingying, and X. Zheng. (2018). “Approaching Mean-Variance Efficiency for Large Portfolios”. *The Review of Financial Studies*. 32(7): 2890–2919.

Arlot, S. and A. Celisse. (2010). “A survey of cross-validation procedures for model selection”. *Statistics surveys*. 4: 40–79.

Aubry, M., R. Kraussl, M. Gustavo, and C. Spaenjers. (2022). “Biased Auctioneers”. *The Journal of Finance, forthcoming*.

Avramov, D., S. Cheng, and L. Metzker. (2022a). “Machine Learning vs. Economic Restrictions: Evidence from Stock Return Predictability”. *Management Science, forthcoming.*

Avramov, D., S. Cheng, L. Metzker, and S. Voigt. (2021). “Integrating factor models”. *Journal of Finance, forthcoming.*

Avramov, D., G. Kaplanski, and A. Subrahmanyam. (2022b). “Postfundamentals Price Drift in Capital Markets: A Regression Regularization Perspective”. *Management Science*. 68(10): 7658–7681.

Avramov, D. and G. Zhou. (2010). “Bayesian Portfolio Analysis”. *Annual Review of Financial Economics*. 2(1): 25–47.

Bai, J. (2003). “Inferential theory for factor models of large dimensions”. *Econometrica*. 71(1): 135–171.

Bai, J. and S. Ng. (2002). “Determining the Number of Factors in Approximate Factor Models”. *Econometrica*. 70: 191–221.

Bai, J. and S. Ng. (2021). “Approximate Factor Models with Weaker Loading”. *Tech. rep.* Columbia University.

Bajgrowicz, P. and O. Scaillet. (2012). “Technical trading revisited: False discoveries, persistence tests, and transaction costs”. *Journal of Financial Economics*. 106(3): 473–491.

Baker, M. and J. Wurgler. (2006). “Investor sentiment and the cross-section of stock returns”. *The journal of Finance*. 61(4): 1645–1680.

Baker, M. and J. Wurgler. (2007). “Investor Sentiment in the Stock Market”. *Journal of Economic Perspectives*. 21(2): 129–152.

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

Bali, T. G., A. Goyal, D. Huang, F. Jiang, and Q. Wen. (2020). “Predicting Corporate Bond Returns: Merton Meets Machine Learning”. *Tech. rep.* Georgetown University.

Bansal, R. and S. Viswanathan. (1993). “No Arbitrage and Arbitrage Pricing: A New Approach”. *The Journal of Finance*. 48(4): 1231–1262.

Bansal, R. and A. Yaron. (2004). “Risks for the long run: A potential resolution of asset pricing puzzles”. *The journal of Finance*. 59(4): 1481–1509.

Bao, W., J. Yue, and Y. Rao. (2017). “A deep learning framework for financial time series using stacked autoencoders and long-short term memory”. *PLOS ONE*. 12(7): 1–24.

Barberis, N. (2018). “Psychology-based models of asset prices and trading volume”. In: *Handbook of behavioral economics: applications and foundations 1*. Vol. 1. Elsevier. 79–175.

Barberis, N. and R. Thaler. (2003). “A survey of behavioral finance”. *Handbook of the Economics of Finance*. 1: 1053–1128.

Barndorff-Nielsen, O. E. and N. Shephard. (2002). “Econometric Analysis of Realized Volatility and Its Use in Estimating Stochastic Volatility Models”. *Journal of the Royal Statistical Society, B*. 64: 253–280.

Barras, L., O. Scaillet, and R. Wermers. (2010). “False discoveries in mutual fund performance: Measuring luck in estimated alphas”. *Journal of Finance*. 65(1): 179–216.

Bartlett, P. L., P. M. Long, G. Lugosi, and A. Tsigler. (2020). “Benign overfitting in linear regression”. *Proceedings of the National Academy of Sciences*. 117(48): 30063–30070.

Basu, S. (1977). “Investment Performance of Common Stocks in Relation to Their Price-Earnings Ratios: A Test of the Efficient Market Hypothesis”. *The Journal of Finance*. 32(3): 663–682.

Belkin, M., D. Hsu, S. Ma, and S. Mandal. (2018). “Reconciling modern machine learning and the biasvariance trade-off. arXiv e-prints”.

Belkin, M. (2021). “Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation”. *Acta Numerica*. 30: 203–248.

Belkin, M., D. Hsu, and J. Xu. (2020). “Two models of double descent for weak features”. *SIAM Journal on Mathematics of Data Science*. 2(4): 1167–1180.

Belkin, M., A. Rakhlin, and A. B. Tsybakov. (2019). “Does data interpolation contradict statistical optimality?” In: *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR. 1611–1619.

Benjamini, Y. and Y. Hochberg. (1995). “Controlling the false discovery rate: a practical and powerful approach to multiple testing”. *Journal of the Royal Statistical Society. Series B (Methodological).* 57(1): 289–300.

Bianchi, D., M. Büchner, and A. Tamoni. (2021). “Bond risk premiums with machine learning”. *The Review of Financial Studies.* 34(2): 1046–1089.

Bickel, P. J. and E. Levina. (2008a). “Covariance Regularization by Thresholding”. *Annals of Statistics.* 36(6): 2577–2604.

Bickel, P. J. and E. Levina. (2008b). “Regularized Estimation of Large Covariance Matrices”. *Annals of Statistics.* 36: 199–227.

Black, F. and R. Litterman. (1992). “Global Portfolio Optimization”. *Financial Analysts Journal.* 48(5): 28–43.

Bollerslev, T., S. Z. Li, and V. Todorov. (2016). “Roughing up beta: Continuous versus discontinuous betas and the cross section of expected stock returns”. *Journal of Financial Economics.* 120: 464–490.

Bollerslev, T., M. C. Medeiros, A. Patton, and R. Quaedvlieg. (2022). “From Zero to Hero: Realized Partial (Co)variances”. *Journal of Econometrics.* 231: 348–360.

Bollerslev, T. and V. Todorov. (2011). “Estimation of jump tails”. *Econometrica.* 79(6): 1727–1783.

Box, G. E. P., G. M. Jenkins, G. C. Reinsel, and G. M. Ljung. (2015). *Time Series Analysis: Forecasting and Control.* 5th. Wiley.

Box, G. E. and G. Jenkins. (1970). *Time Series Analysis: Forecasting and Control.* San Francisco: Holden-Day.

Brandt, M. W. (1999). “Estimating Portfolio and Consumption Choice: A Conditional Euler Equations Approach”. *The Journal of Finance.* 54(5): 1609–1645.

Brandt, M. W. (2010). “Portfolio Choice Problems”. In: *Handbook of Financial Econometrics.* Ed. by Y. Ait-Sahalia and L. P. Hansen. Amsterdam, The Netherlands: North-Holland. 269–336.

Brandt, M. W. and P. Santa-Clara. (2006). “Dynamic Portfolio Selection by Augmenting the Asset Space”. *The Journal of Finance.* 61(5): 2187–2217.

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

Breiman, L. (1995). "The Mathematics of Generalization". In: CRC Press. Chap. Reflections After Refereeing Papers for NIPS. 11–15.

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

Britten-Jones, M. (1999). "The Sampling Error in Estimates of Mean-Variance Efficient Portfolio Weights". *The Journal of Finance*. 54(2): 655–671.

Brown, T., B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. (2020). "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc. 1877–1901.

Bryzgalova, S., V. DeMiguel, S. Li, and M. Pelger. (2023). "Asset-Pricing Factors with Economic Targets". *Available at SSRN 4344837*.

Bryzgalova, S., M. Pelger, and J. Zhu. (2020). "Forest through the Trees: Building Cross-Sections of Asset Returns". *Tech. rep.* London School of Business and Stanford University.

Büchner, M. and B. T. Kelly. (2022). "A factor model for option returns". *Journal of Financial Economics*.

Bybee, L., L. Gomes, and J. Valente. (2023a). "Macro-based factors for the cross-section of currency returns".

Bybee, L., B. T. Kelly, A. Manela, and D. Xiu. (2020). "The structure of economic news". *Tech. rep.* National Bureau of Economic Research.

Bybee, L., B. T. Kelly, and Y. Su. (2023b). "Narrative asset pricing: Interpretable systematic risk factors from news text". *Review of Financial Studies*.

Cai, T. and W. Liu. (2011). "Adaptive Thresholding for Sparse Covariance Matrix Estimation". *Journal of the American Statistical Association*. 106: 672–684.

Campbell, J. Y. and J. H. Cochrane. (1999). “By force of habit: A consumption-based explanation of aggregate stock market behavior”. *Journal of political Economy*. 107(2): 205–251.

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

Campbell, J. Y. and R. J. Shiller. (1988). “Stock Prices, Earnings, and Expected Dividends”. *The Journal of Finance*. 43(3): 661–676.

Chamberlain, G. and M. Rothschild. (1983). “Arbitrage, Factor Structure, and Mean-Variance Analysis on Large Asset Markets”. *Econometrica*. 51: 1281–1304.

Chatelais, N., A. Stalla-Bourdillon, and M. D. Chinn. (2023). “Forecasting real activity using cross-sectoral stock market information”. *Journal of International Money and Finance*. 131: 102800.

Chen, A. Y. and T. Zimmermann. (2021). “Open Source Cross-Sectional Asset Pricing”. *Critical Finance Review, Forthcoming*.

Chen, B., Q. Yu, and G. Zhou. (2023). “Useful factors are fewer than you think”. *Available at SSRN 3723126*.

Chen, H., W. W. Dou, and L. Kogan. (2022a). “Measuring “Dark Matter” in Asset Pricing Models”. *Journal of Finance, forthcoming*.

Chen, J., G. Tang, J. Yao, and G. Zhou. (2022b). “Investor Attention and Stock Returns”. *Journal of Financial and Quantitative Analysis*. 57(2): 455–484.

Chen, L., M. Pelger, and J. Zhu. (2021). “Deep learning in asset pricing”. *SSRN*.

Chen, X. and S. C. Ludvigson. (2009). “Land of addicts? an empirical investigation of habit-based asset pricing models”. *Journal of Applied Econometrics*. 24(7): 1057–1093.

Chernozhukov, V., D. Chetverikov, M. Demirer, E. Duflo, C. Hansen, W. K. Newey, and J. Robins. (2018). “Double/debiased Machine Learning for Treatment and Structure Parameters”. *The Econometrics Journal*. 21(1): C1–C68.

Chib, S., L. Zhao, and G. Zhou. (2023). “Winners from winners: A tale of risk factors”. *Management Science*.

Chinco, A., A. D. Clark-Joseph, and M. Ye. (2019). “Sparse Signals in the Cross-Section of Returns”. *Journal of Finance*. 74(1): 449–492.

Cho, K., B. van Merriënboer, D. Bahdanau, and Y. Bengio. (2014). "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches". In: *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation.* 103–111. DOI: 10.3115/v1/W14-4012. URL: https://aclanthology.org/W14-4012.

Choi, D., W. Jiang, and C. Zhang. (2022). "Alpha Go Everywhere: Machine Learning and International Stock Returns". *Tech. rep.* The Chinese University of Hong Kong.

Chong, E., C. Han, and F. C. Park. (2017). "Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies". *Expert Systems with Applications.* 83: 187–205.

Cochrane, J. H. (2009). *Asset pricing: Revised edition.* Princeton university press.

Cochrane, J. H. (2008). "The Dog That Did Not Bark: A Defense of Return Predictability". *The Review of Financial Studies.* 21(4): 1533–1575.

Cochrane, J. H. and M. Piazzesi. (2005). "Bond Risk Premia". *American Economic Review.* 95(1): 138–160.

Cong, L. W., G. Feng, J. He, and X. He. (2022). "Asset Pricing with Panel Tree Under Global Split Criteria". *Tech. rep.* City University of Hong Kong.

Cong, L. W., K. Tang, J. Wang, and Y. Zhang. (2020). "AlphaPortfolio for Investment and Economically Interpretable AI". *Available at SSRN.*

Connor, G., M. Hagmann, and O. Linton. (2012). "Efficient semiparametric estimation of the Fama–French model and extensions". *Econometrica.* 80(2): 713–754.

Connor, G. and R. A. Korajczyk. (1986). "Performance Measurement with the Arbitrage Pricing Theory: A New Framework for Analysis". *Journal of Financial Economics.* 15(3): 373–394.

Connor, G. and R. A. Korajczyk. (1988). "Risk and return in an equilibrium APT: Application of a new test methodology". *Journal of financial economics.* 21(2): 255–289.

Correia, M., J. Kang, and S. Richardson. (2018). "Asset volatility". *Review of Accounting Studies.* 23(1): 37–94.

Corsi, F. (2009). “A simple approximate long-memory model of realized volatility”. *Journal of Financial Econometrics*. 7: 174–196.

Cowles, A. 3. (1933). “Can Stock Market Forecasters Forecast?” *Econometrica*. 1(3): 309–324.

Cujean, J. and M. Hasler. (2017). “Why Does Return Predictability Concentrate in Bad Times?” *The Journal of Finance*. 72(6): 2717–2758.

Cybenko, G. (1989). “Approximation by superpositions of a sigmoidal function”. *Mathematics of control, signals and systems*. 2(4): 303–314.

Da, R., S. Nagel, and D. Xiu. (2022). “The Statistical Limit of Arbitrage”. *Tech. rep.* Chicago Booth.

Das, S. R. *et al.* (2014). “Text and context: Language analytics in finance”. *Foundations and Trends® in Finance*. 8(3): 145–261.

Davis, S. J., S. Hansen, and C. Seminario-Amez. (2020). “Firm-Level Risk Exposures and Stock Returns in the Wake of COVID-19”. *Tech. rep.* University of Chicago.

DeMiguel, V., A. Martin-Utrera, F. J. Nogales, and R. Uppal. (2020). “A Transaction-Cost Perspective on the Multitude of Firm Characeristics”. *The Review of Financial Studies*. 33(5): 2180–2222.

Deng, W., L. Gao, B. Hu, and G. Zhou. (2022). “Seeing is Believing: Annual Report”. *Available at SSRN 3723126*.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova. (2018). “Bert: Pre-training of deep bidirectional transformers for language understanding”. *arXiv preprint arXiv:1810.04805*.

Didisheim, A., S. Ke, B. Kelly, and S. Malamud. (2023). “Complexity in Factor Pricing Models”. *Tech. rep.* Yale University.

Easley, D., M. López de Prado, M. O’Hara, and Z. Zhang. (2020). “Microstructure in the Machine Age”. *The Review of Financial Studies*. 34(7): 3316–3363.

Erel, I., L. H. Stern, C. Tan, and M. S. Weisbach. (2021). “Selecting directors using machine learning”. *The Review of Financial Studies*. 34(7): 3226–3264.

Fabozzi, F. J., D. Huang, and G. Zhou. (2010). “Robust portfolios: contributions from operations research and finance”. *Annals of Operations Research*. 176(1): 191–220.

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.

Fama, E. F. and K. R. French. (2010). “Luck versus skill in the cross-section of mutual fund returns”. *The Journal of Finance*. 65(5): 1915–1947.

Fama, E. F. and K. R. French. (2015). “A five-factor asset pricing model”. *Journal of financial economics*. 116(1): 1–22.

Fama, E. F. (1970). “Efficient Capital Markets: A Review of Theory and Empirical Work”. *The Journal of Finance*. 25(2): 383–417.

Fama, E. F. (1990). “Stock Returns, Expected Returns, and Real Activity”. *The Journal of Finance*. 45(4): 1089–1108.

Fama, E. F. and R. R. Bliss. (1987). “The Information in Long-Maturity Forward Rates”. *The American Economic Review*. 77(4): 680–692.

Fama, E. F. and K. R. French. (1992). “The Cross-Section of Expected Stock Returns”. *The Journal of Finance*. 47: 427–465.

Fama, E. F. and J. D. Macbeth. (1973). “Risk, Return, and Equilibrium: Empirical Tests”. *Journal of Political Economy*. 81(3): 607–636.

Fan, J., Y. Fan, and J. Lv. (2008). “High Dimensional Covariance Matrix Estimation using a Factor Model”. *Journal of Econometrics*. 147: 186–197.

Fan, J., A. Furger, and D. Xiu. (2016a). “Incorporating Global Industrial Classification Standard into Portfolio Allocation: A Simple Factor-Based Large Covariance Matrix Estimator with High Frequency Data”. *Journal of Business and Economic Statistics*. 34(4): 489–503.

Fan, J., Y. Liao, and M. Mincheva. (2013). “Large Covariance Estimation by Thresholding Principal Orthogonal Complements”. *Journal of the Royal Statistical Society, B*. 75: 603–680.

Fan, J., Y. Liao, and W. Wang. (2016b). “Projected principal component analysis in factor models”. *Annals of Statistics*. 44(1): 219.

Fan, J., Y. Liao, and J. Yao. (2015). “Power Enhancement in High-Dimensional Cross-Sectional Tests”. *Econometrica*. 83(4): 14977–1541.

Feng, G., S. Giglio, and D. Xiu. (2020). “Taming the Factor Zoo: A Test of New Factors”. *Journal of Finance*. 75(3): 1327–1370.

Feng, G., J. He, and N. G. Polson. (2018). “Deep learning for predicting asset returns”. *arXiv preprint arXiv:1804.09314*.

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

Friedman, J. H. (2001). “Greedy function approximation: a gradient boosting machine”. *Annals of statistics*: 1189–1232.

Frost, P. A. and J. E. Savarino. (1986). “An Empirical Bayes Approach to Efficient Portfolio Selection”. *The Journal of Financial and Quantitative Analysis*. 21(3): 293–305. (Accessed on 01/30/2023).

Fuster, A., P. Goldsmith-Pinkham, T. Ramadorai, and A. Walther. (2022). “Predictably unequal? The effects of machine learning on credit markets”. *The Journal of Finance*. 77(1): 5–47.

Gabaix, X. (2012). “Variable rare disasters: An exactly solved framework for ten puzzles in macro-finance”. *The Quarterly Journal of Economics*. 127: 645–700.

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

Gagliardini, P. and D. Ronchetti. (2019). “Comparing Asset Pricing Models by the Conditional Hansen-Jagannathan Distance*”. *Journal of Financial Econometrics*. 18(2): 333–394.

Garcia, D., X. Hu, and M. Rohrer. (2022). “The colour of finance words”. *Tech. rep.* University of Colorado at Boulder.

Garleanu, N. and L. H. Pedersen. (2013). “Dynamic Trading with Predictable Returns and Transaction Costs”. *The Journal of Finance*. 68(6): 2309–2340.

Gentzkow, M., B. Kelly, and M. Taddy. (2019). “Text as data”. *Journal of Economic Literature*. 57(3): 535–74.

Geweke, J. and G. Zhou. (1996). “Measuring the pricing error of the arbitrage pricing theory”. *The review of financial studies*. 9(2): 557–587.

Gibbons, M. R., S. A. Ross, and J. Shanken. (1989). “A test of the efficiency of a given portfolio”. *Econometrica: Journal of the Econometric Society*: 1121–1152.

Giglio, S., B. Kellly, and D. Xiu. (2022a). “Factor Models, Machine Learning, and Asset Pricing”. *Annual Review of Financial Economics.* 14: 1–32.

Giglio, S., Y. Liao, and D. Xiu. (2021a). “Thousands of Alpha Tests”. *Review of Financial Studies.* 34(7): 3456–3496.

Giglio, S. and D. Xiu. (2021). “Asset Pricing with Omitted Factors”. *Journal of Political Economy.* 129(7): 1947–1990.

Giglio, S., D. Xiu, and D. Zhang. (2021b). “Test Assets and Weak Factors”. *Tech. rep.* Yale University and University of Chicago.

Giglio, S., D. Xiu, and D. Zhang. (2022b). “Prediction when Factors are Weak”. *Tech. rep.* Yale University and University of Chicago.

Glaeser, E. L., M. S. Kincaid, and N. Naik. (2018). “Computer Vision and Real Estate: Do Looks Matter and Do Incentives Determine Looks”. *Tech. rep.* Harvard University.

Goodfellow, I., Y. Bengio, and A. Courville. (2016). *Deep learning.* MIT press.

Goulet Coulombe, P. and M. Göbel. (2023). “Maximally Machine-Learnable Portfolios”. *Available at SSRN 4428178.*

Goyal, A. and A. Saretto. (2022). “Are Equity Option Returns Abnormal? IPCA Says No”. *IPCA Says No (August 19, 2022).*

Gu, S., B. Kelly, and D. Xiu. (2020a). “Autoencoder Asset Pricing Models”. *Journal of Econometrics.*

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

Guijarro-Ordonez, J., M. Pelger, and G. Zanotti. (2022). “Deep Learning Statistical Arbitrage”. *Tech. rep.* Stanford University.

Hambly, B., R. Xu, and H. Yang. (2022). “Recent Advances in Reinforcement Learning in Finance”. *Tech. rep.* University of Oxford.

Hansen, L. P. and R. Jagannathan. (1997). “Assessing Specification Errors in Stochastic Discount Factor Models”. *Journal of Finance.* 52: 557–590.

Hansen, L. P. and S. F. Richard. (1987). “The role of conditioning information in deducing testable restrictions implied by dynamic asset pricing models”. *Econometrica: Journal of the Econometric Society*: 587–613.

Hansen, L. P. and K. J. Singleton. (1982). "Generalized Instrumental Variables Estimation of Nonlinear Rational Expectations Models". *Econometrica*. 50(5): 1269–1286.

Harvey, C. R. and Y. Liu. (2020). "False (and missed) discoveries in financial economics". *Journal of Finance*. 75(5): 2503–2553.

Harvey, C. R., Y. Liu, and H. Zhu. (2016). "... And the cross-section of expected returns". *Review of Financial Studies*. 29(1): 5–68.

Harvey, C. R. (2017). "Presidential Address: The Scientific Outlook in Financial Economics". *Journal of Finance*. 72(4): 1399–1440.

Harvey, C. R. and W. E. Ferson. (1999). "Conditioning Variables and the Cross-Section of Stock Returns". *Journal of Finance*. 54: 1325–1360.

Hastie, T., A. Montanari, S. Rosset, and R. J. Tibshirani. (2019). "Surprises in high-dimensional ridgeless least squares interpolation". *arXiv preprint arXiv:1903.08560*.

Haugen, R. A. and N. L. Baker. (1996). "Commonality in the determinants of expected stock returns". *Journal of Financial Economics*. 41(3): 401–439.

Hayek, F. A. (1945). "The Use of Knowledge in Society". *The American Economic Review*. 35(4): 519–530.

He, A., S. He, D. Rapach, and G. Zhou. (2022a). "Expected Stock Returns in the Cross-section: An Ensemble Approach". *Working Paper*.

He, K., X. Zhang, S. Ren, and J. Sun. (2016). "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778. DOI: 10.1109/CVPR.2016.90.

He, S., M. Yuan, and G. Zhou. (2022b). "Principal Portfolios: A Note". *Working Paper*.

He, X., G. Feng, J. Wang, and C. Wu. (2021). "Predicting Individual Corporate Bond Returns". *Tech. rep.* City University of Hong Kong.

He, Z., B. Kelly, and A. Manela. (2017). "Intermediary asset pricing: New evidence from many asset classes". *Journal of Financial Economics*. 126(1): 1–35.

Hochreiter, S. and J. Schmidhuber. (1997). "Long short-term memory". *Neural Computation*. 9: 1735–1780.

Hong, H., T. Lim, and J. C. Stein. (2000). "Bad News Travels Slowly: Size, Analyst Coverage, and the Profitability of Momentum Strategies". *The Journal of Finance*. 55(1): 265–295.

Hornik, K., M. Stinchcombe, and H. White. (1989). "Multilayer feedforward networks are universal approximators". *Neural networks*. 2(5): 359–366.

Hornik, K., M. Stinchcombe, and H. White. (1990). "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks". *Neural networks*. 3(5): 551–560.

Hou, K., C. Xue, and L. Zhang. (2018). "Replicating Anomalies". *The Review of Financial Studies*. 33(5): 2019–2133.

Hu, A. and S. Ma. (2020). "Human interactions and financial investment: A video-based approach". *Available at SSRN*.

Huang, D., F. Jiang, K. Li, G. Tong, and G. Zhou. (2021). "Scaled PCA: A New Approach to Dimension Reduction". *Management Science, forthcoming*.

Huang, D., F. Jiang, J. Tu, and G. Zhou. (2014). "Investor Sentiment Aligned: A Powerful Predictor of Stock Returns". *The Review of Financial Studies*. 28(3): 791–837. ISSN: 0893-9454.

Huang, J., J. L. Horowitz, and F. Wei. (2010). "Variable selection in nonparametric additive models". *The Annals of Statistics*. 38(4): 2282–2313.

Huberman, G. (1982). "A Simple Approach to Arbitrage Pricing Theory". *Journal of Economic Thoery*. 28(1): 183–191.

Ingersoll, J. E. (1984). "Some Results in the Theory of Arbitrage Pricing". *Journal of Finance*. 39(4): 1021–1039.

Israel, R., B. Kellly, and T. J. Moskowitz. (2020). "Can Machines "Learn" Finance?" *Journal of Investment Management*. 18(2): 23–36.

Israelov, R. and B. T. Kelly. (2017). "Forecasting the distribution of option returns". *Available at SSRN 3033242*.

Jacot, A., F. Gabriel, and C. Hongler. (2018). "Neural tangent kernel: Convergence and generalization in neural networks". *arXiv preprint arXiv:1806.07572*.

Jegadeesh, N. and D. Wu. (2013). "Word power: A new approach for content analysis". *Journal of Financial Economics*. 110(3): 712–729.

Jensen, T. I., B. Kellly, C. Seminario-Amez, and L. H. Pedersen. (2022). "Machine Learning and the Implementable Efficient Frontier". *Tech. rep.* Copenhagen Business School.

Jensen, T. I., B. Kelly, and L. H. Pedersen. (2021). "Is There a Replication Crisis in Finance?" *Journal of Finance, forthcoming.*

Jiang, F., G. Tang, and G. Zhou. (2018). "Firm characteristics and Chinese stocks". *Journal of Management Science and Engineering.* 3(4): 259–283.

Jiang, J., B. Kelly, and D. Xiu. (2022). "(Re-)Imag(in)ing Price Trends". *Journal of Finance, forthcoming.*

Jiang, J., B. Kelly, and D. Xiu. (2023). "Expected Returns and Large Language Models". *Tech. rep.* University of Chicago and Yale University.

Jobson, J. D. and B. Korkie. (1980). "Estimation for Markowitz Efficient Portfolios". *Journal of the American Statistical Association.* 75(371): 544–554. (Accessed on 01/30/2023).

Jorion, P. (1986). "Bayes-Stein Estimation for Portfolio Analysis". *The Journal of Financial and Quantitative Analysis.* 21(3): 279–292. (Accessed on 01/30/2023).

Jurado, K., S. C. Ludvigson, and S. Ng. (2015). "Measuring uncertainty". *The American Economic Review.* 105(3): 1177–1216.

Kan, R., X. Wang, and G. Zhou. (2022). "Optimal Portfolio Choice with Estimation Risk: No Risk-free Asset Case". *Management Science, forthcoming.*

Kan, R. and C. Zhang. (1999). "Two-Pass Tests of Asset Pricing Models with Useless Factors". *The Journal of Finance.* 54(1): 203–235.

Kan, R. and G. Zhou. (2007). "Optimal Portfolio Choice with Parameter Uncertainty". *Journal of Financial and Quantitative Analysis.* 42(3): 621–656.

Ke, T., B. Kelly, and D. Xiu. (2019). "Predicting Returns with Text Data". *Tech. rep.* Harvard University, Yale University, and the University of Chicago.

Kelly, B., S. Malamud, and L. H. Pedersen. (2020a). "Principal Portfolios". *Working Paper.*

Kelly, B., S. Malamud, and K. Zhou. (2022a). "Virtue of Complexity in Return Prediction". *Tech. rep.* Yale University.

Kelly, B., A. Manela, and A. Moreira. (2018). "Text Selection". *Working paper.*

Kelly, B., T. Moskowitz, and S. Pruitt. (2021). "Understanding Momentum and Reversal". *Journal of Financial Economics.* 140(3): 726–743.

Kelly, B., D. Palhares, and S. Pruitt. (Forthcoming). "Modeling Corporate Bond Returns". *Journal of Finance.*

Kelly, B. and S. Pruitt. (2013). "Market expectations in the cross-section of present values". *The Journal of Finance.* 68(5): 1721–1756.

Kelly, B. and S. Pruitt. (2015). "The three-pass regression filter: A new approach to forecasting using many predictors". *Journal of Econometrics.* 186(2): 294–316. ISSN: 0304-4076.

Kelly, B., S. Pruitt, and Y. Su. (2020b). "Characteristics are Covariances: A Unified Model of Risk and Return". *Journal of Financial Economics.*

Kelly, B. T., S. Malamud, and K. Zhou. (2022b). "The Virtue of Complexity Everywhere". *Available at SSRN.*

Kim, S., R. Korajczyk, and A. Neuhierl. (2020). "Arbitrage Portfolios". *Review of Financial Studies, forthcoming.*

Koopmans, T. C. (1947). "Measurement without theory". *The Review of Economics and Statistics.* 29(3): 161–172.

Kosowski, R., A. Timmermann, R. Wermers, and H. White. (2006). "Can mutual fund "stars" really pick stocks? New evidence from a bootstrap analysis". *The Journal of Finance.* 61(6): 2551–2595.

Kozak, S. (2020). "Kernel trick for the cross-section". *Available at SSRN 3307895.*

Kozak, S., S. Nagel, and S. Santosh. (2018). "Interpreting factor models". *The Journal of Finance.* 73(3): 1183–1223.

Kozak, S., S. Nagel, and S. Santosh. (2020). "Shrinking the cross-section". *Journal of Financial Economics.* 135(2): 271–292.

Ledoit, O. and M. Wolf. (2004). "Honey, I shrunk the sample covariance matrix". *Journal of Portfolio Management.* 30: 110–119.

Ledoit, O. and M. Wolf. (2012). "Nonlinear shrinkage estimation of large-dimensional covariance matrices". *The Annals of Statistics.* 40: 1024–1060.

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

Lettau, M. and S. Ludvigson. (2001). “Consumption, Aggregate Wealth, and Expected Stock Returns”. *The Journal of Finance*. 56(3): 815–849.

Lettau, M. and M. Pelger. (2020a). “Estimating Latent Asset-Pricing Factors”. *Journal of Econometrics*. 218: 1–31.

Lettau, M. and M. Pelger. (2020b). “Factors that fit the time series and cross-section of stock returns”. *Review of Financial Studies*. 33(5): 2274–2325.

Lewellen, J. (2015). “The Cross-section of Expected Stock Returns”. *Critical Finance Review*. 4(1): 1–44.

Li, K., F. Mai, R. Shen, and X. Yan. (2021). “Measuring corporate culture using machine learning”. *The Review of Financial Studies*. 34(7): 3265–3315.

Li, S. Z. and Y. Tang. (2022). “Automated Risk Forecasting”. *Tech. rep.* Rutgers, The State University of New Jersey.

Light, N., D. Maslov, and O. Rytchkov. (2017). “Aggregation of Information About the Cross Section of Stock Returns: A Latent Variable Approach”. *The Review of Financial Studies*. 30(4): 1339–1381.

Lo, A. W. and A. C. MacKinlay. (1990). “Data-snooping biases in tests of financial asset pricing models”. *Review of financial studies*. 3(3): 431–467.

Loughran, T. and B. McDonald. (2011). “When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks”. *The Journal of Finance*. 66(1): 35–65.

Loughran, T. and B. McDonald. (2020). “Textual analysis in finance”. *Annual Review of Financial Economics*. 12: 357–375.

Lucas Jr, R. E. (1976). “Econometric policy evaluation: A critique”. In: *Carnegie-Rochester conference series on public policy*. Vol. 1. North-Holland. 19–46.

Ludvigson, S. C. and S. Ng. (2010). “A factor analysis of bond risk premia”. In: *Handbook of empirical economics and finance*. Ed. by A. Ulah and D. E. A. Giles. Vol. 1. Chapman and Hall, Boca Raton, FL. Chap. 12. 313–372.

Ludvigson, S. C. (2013). “Chapter 12 - Advances in Consumption-Based Asset Pricing: Empirical Tests”. In: ed. by G. M. Constantinides, M. Harris, and R. M. Stulz. Vol. 2. *Handbook of the Economics of Finance*. Elsevier. 799–906.

Ludvigson, S. C. and S. Ng. (2007). “The empirical risk–return relation: A factor analysis approach”. *Journal of Financial Economics*. 83(1): 171–222.

Lynch, A. W. and P. Balduzzi. (2000). “Predictability and Transaction Costs: The Impact on Rebalancing Rules and Behavior”. *The Journal of Finance*. 55(5): 2285–2309.

Lyonnet, V. and L. H. Stern. (2022). “Venture Capital (Mis) allocation in the Age of AI”. *Available at SSRN 4260882*.

Malloy, C. J., T. J. Moskowitz, and A. Vissing-Jorgensen. (2009). “Long-run stockholder consumption risk and asset returns”. *The Journal of Finance*. 64(6): 2427–2479.

Manela, A. and A. Moreira. (2017). “News implied volatility and disaster concerns”. *Journal of Financial Economics*. 123(1): 137–162.

Markowitz, H. (1952). “Portfolio selection”. *Journal of Finance*. 7(1): 77–91.

Martin, I. W. and S. Nagel. (2021). “Market efficiency in the age of big data”. *Journal of Financial Economics*.

Mehra, R. and E. C. Prescott. (1985). “The equity premium: A puzzle”. *Journal of Monetary Economics*. 15(2): 145–161.

Menzly, L., T. Santos, and P. Veronesi. (2004). “Understanding Predictability”. *Journal of Political Economy*. 112(1): 1–47. (Accessed on 02/06/2023).

Merton, R. C. (1973). “An Intertemporal Capital Asset Pricing Model”. *Econometrica*. 41: 867–887.

Michaud, R. O. (1989). “The Markowitz Optimization Enigma: Is 'Optimized' Optimal?” *Financial Analysts Journal*. 45(1): 31–42. (Accessed on 01/30/2023).

Mittnik, S., N. Robinzonov, and M. Spindler. (2015). “Stock market volatility: Identifying major drivers and the nature of their impact”. *Journal of Banking & Finance*. 58: 1–14.

Moritz, B. and T. Zimmermann. (2016). “Tree-Based Conditional Portfolio Sorts: The Relation Between Past and Future Stock Returns”. *Tech. rep.* Ludwig Maximilian University Munich.

Nagel, S. and K. Singleton. (2011). “Estimation and Evaluation of Conditional Asset Pricing Models”. *The Journal of Finance.* 66(3): 873–909. (Accessed on 02/20/2023).

Nishii, R. (1984). “Asymptotic Properties of Criteria for Selection of Variables in Multiple Regression”. *The Annals of Statistics.* 12(2): 758–765.

Novy-Marx, R. (2014). “Predicting anomaly performance with politics, the weather, global warming, sunspots, and the stars”. *Journal of Financial Economics.* 112(2): 137–146.

Obaid, K. and K. Pukthuanthong. (2022). “A picture is worth a thousand words: Measuring investor sentiment by combining machine learning and photos from news”. *Journal of Financial Economics.* 144: 273–297.

Ohlson, J. A. (1980). “Financial Ratios and the Probabilistic Prediction of Bankruptcy”. *Journal of Accounting Research.* 18(1): 109–131.

Onatski, A. (2009). “Testing hypotheses about the number of factors in large factor models”. *Econometrica.* 77(5): 1447–1479.

Onatski, A. (2010). “Determining the Number of Factors from Empirical Distribution of Eigenvalues”. *Review of Economics and Statistics.* 92: 1004–1016.

Onatski, A. (2012). “Asymptotics of the principal components estimator of large factor models with weakly influential factors”. *Journal of Econometrics.* 168: 244–258.

Pastor, L. (2000). “Portfolio Selection and Asset Pricing Models”. *The Journal of Finance.* 55(1): 179–223.

Pastor, L. and R. F. Stambaugh. (2000). “Comparing Asset Pricing Models: An Investment Perspective”. *Journal of Financial Economics.* 56: 335–381.

Pástor, L. and R. F. Stambaugh. (2003). “Liquidity Risk and Expected Stock Returns”. *Journal of Political Economy.* 111(3): 642–685.

Pesaran, H. and T. Yamagata. (2017). “Testing for Alpha in Linear Factor Pricing Models with a Large Number of Securities”. *Tech. rep.*

Petersen, M. A. (2008). “Estimating Standard Errors in Finance Panel Data Sets: Comparing Approaches”. *The Review of Financial Studies*. 22(1): 435–480.

Pukthuanthong, K., R. Roll, and A. Subrahmanyam. (2019). “A Protocol for Factor Identification”. *Review of Financial Studies*. 32(4): 1573–1607.

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.* (2019). “Language models are unsupervised multitask learners”. *OpenAI blog*. 1(8): 9.

Rahimi, A. and B. Recht. (2007). “Random Features for Large-Scale Kernel Machines.” In: *NIPS*. Vol. 3. No. 4. Citeseer. 5.

Rapach, D. and G. Zhou. (2013). “Chapter 6 - Forecasting Stock Returns”. In: *Handbook of Economic Forecasting*. Ed. by G. Elliott and A. Timmermann. Vol. 2. *Handbook of Economic Forecasting*. Elsevier. 328–383.

Rapach, D. and G. Zhou. (2022). “Asset pricing: Time-series predictability”.

Rapach, D. E., J. K. Strauss, and G. Zhou. (2010). “Out-of-sample equity premium prediction: Combination forecasts and links to the real economy”. *The Review of Financial Studies*. 23(2): 821–862.

Rapach, D. E., J. K. Strauss, and G. Zhou. (2013). “International stock return predictability: what is the role of the United States?” *The Journal of Finance*. 68(4): 1633–1662.

Rather, A. M., A. Agarwal, and V. Sastry. (2015). “Recurrent neural network and a hybrid model for prediction of stock returns”. *Expert Systems with Applications*. 42(6): 3234–3241.

Roll, R. (1977). “A Critique of the Asset Pricing Theory’s Tests”. *Journal of Financial Economics*. 4: 129–176.

Rosenberg, B. (1974). “Extra-Market Components of Covariance in Security Returns”. *Journal of Financial and Quantitative Analysis*. 9(2): 263–274.

Rosenberg, J. V. and R. F. Engle. (2002). “Empirical pricing kernels”. *Journal of Financial Economics*. 64(3): 341–372.

Ross, S. A. (1976). “The Arbitrage Theory of Capital Asset Pricing”. *Journal of Economic Theory*. 13(3): 341–360.

Rossi, A. G. (2018). "Predicting stock market returns with machine learning". *Georgetown University.*

Rossi, A. G. and A. Timmermann. (2015). "Modeling Covariance Risk in Merton's ICAPM". *The Review of Financial Studies.* 28(5): 1428–1461.

Samuelson, P. A. (1965). "Rational Theory of Warrant Pricing". *Industrial Management Review.* 6(2): 13–39.

Schaller, H. and S. V. Norden. (1997). "Regime switching in stock market returns". *Applied Financial Economics.* 7(2): 177–191.

Schapire, R. E. (1990). "The Strength of Weak Learnability". *Machine Learning.* 5(2): 197–227.

Sezer, O. B., M. U. Gudelek, and A. M. Ozbayoglu. (2020). "Financial time series forecasting with deep learning : A systematic literature review: 2005–2019". *Applied Soft Computing.* 90: 106–181.

Shanken, J. (1992a). "On the Estimation of Beta Pricing Models". *Review of Financial Studies.* 5: 1–33.

Shanken, J. (1992b). "The Current State of the Arbitrage Pricing Theory". *Journal of Finance.* 47(4): 1569–1574.

Shiller, R. J. (1981). "Do Stock Prices Move Too Much to be Justified by Subsequent Changes in Dividends?" *The American Economic Review.* 71(3): 421–436.

Simon, F., S. Weibels, and T. Zimmermann. (2022). "Deep Parametric Portfolio Policies". *Tech. rep.* University of Cologne.

Singh, R. and S. Srivastava. (2017). "Stock prediction using deep learning". *Multimedia Tools and Applications.* 76(18): 18569–18584.

Spigler, S., M. Geiger, S. d'Ascoli, L. Sagun, G. Biroli, and M. Wyart. (2019). "A jamming transition from under-to over-parametrization affects generalization in deep learning". *Journal of Physics A: Mathematical and Theoretical.* 52(47): 474001.

Stock, J. H. and M. W. Watson. (2002). "Forecasting Using Principal Components from a Large Number of Predictors". *Journal of the American Statistical Association.* 97(460): 1167–1179.

Stone, M. (1977). "An Asymptotic Equivalence of Choice of Model by Cross-Validation and Akaike's Criterion". *Journal of the Royal Statistical Society Series B.* 39(1): 44–47.

Sullivan, R., A. Timmermann, and H. White. (1999). “Data-snooping, technical trading rule performance, and the bootstrap”. *The journal of Finance*. 54(5): 1647–1691.

Taddy, M. (2013). “Multinomial inverse regression for text analysis”. *Journal of American Statistical Association*. 108(503): 755–770.

Tetlock, P. C. (2007). “Giving Content to Investor Sentiment: The Role of Media in the Stock Market”. *The Journal of Finance*. 62(3): 1139–1168.

Tu, J. and G. Zhou. (2010). “Incorporating Economic Objectives into Bayesian Priors: Portfolio Choice under Parameter Uncertainty”. *The Journal of Financial and Quantitative Analysis*. 45(4): 959–986. (Accessed on 01/30/2023).

Van Binsbergen, J. H. and R. Koijen. (2010). “Predictive Regressions: A Present-Value Approach”. *The Journal of Finance*. 65(4): 1439–1471.

Van Binsbergen, J. H. and C. C. Opp. (2019). “Real anomalies”. *Journal of Finance*. 74(4): 1659–1706.

Wachter, J. (2013). “Can Time-Varying Risk of Rare Disasters Explain Aggregate Stock Market Volatility?” *The Journal of Finance*. 68: 987–1035.

Wachter, J. A. (2006). “A consumption-based model of the term structure of interest rates”. *Journal of Financial Economics*. 79(2): 365–399.

Welch, I. and A. Goyal. (2008). “A comprehensive look at the empirical performance of equity premium prediction”. *The Review of Financial Studies*. 21(4): 1455–1508.

White, H. (2000). “A reality check for data snooping”. *Econometrica*. 68(5): 1097–1126.

Yogo, M. (2006). “A Consumption-Based Explanation of Expected Stock Returns”. *The Journal of Finance*. 61(2): 539–580.

Yuan, M. and G. Zhou. (2022). “Why Naive 1/N Diversification Is Not So Naive, and How to Beat It?” *Available at SSRN*.

Zhang, S., S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer. (2022). "OPT: Open Pre-trained Transformer Language Models". arXiv: 2205.01068 [cs.CL].