Empirical Pricing Kernels and the Permanent-Transitory Decomposition via Machine Learning*

Thesis for the MSc. in Finance at Stockholm School of Economics

Andreas Karlsson†

Supervisor: Prof. Gualtiero Azzalini

May 12, 2024

Abstract

Using a novel dataset of weekly Nordic stocks, I evaluate the asset pricing performance of 3 established machine learning models; a generative adversarial network (GAN), a feedforward neural network (FFN), and an elastic net (EN). The strong performance previously documented for these models translate well to this lower signal-to-noise ratio setting. Although all machine learning models significantly outperform the benchmark CAPM and Fama-French factor models, the GAN stands out. The adversarial estimation procedure allows the GAN to outperform the other models in terms of (i) signal-to-noise ratio, (ii) predictive power, and (iii) explanatory power. The second contribution is to extend the GAN model by imposing the permanent-transitory decomposition of the pricing kernel as an economic constraint. By adding this constraint in an economically informed way, I can produce meaningful improvements in the ability of the GAN to separate the signal from the noise and to explain individual and average returns. This supports a recently established principle in asset pricing that adding economic constraints to machine learning models will allow the models to perform better at asset pricing tasks. I conclude that the machine learning models load heavily on predictable patterns generated by market frictions and limits-to-arbitrage (LTA), and do so in an approximately linear way. While capturing interactions is of first-order importance, other complex nonlinearities are only of second-order importance.

Keywords: Asset pricing, pricing kernel decomposition, stock returns, machine learning, generative adversarial networks, econometrics, GMM.
JEL Classifications: C45, C51, C52, C55, C58, G11, G12

*Acknowledgements: I would like to express gratitude to my supervisor, assistant professor Azzalini, for valuable guidance in writing this thesis. I dedicate this thesis to my nephews Ludvig and Walter.

†Double degree with the University of St. Gallen. Predoctoral Research Fellow, Swedish House of Finance. Email: andreas.karlsson@student.unisg.ch and andreas.karlsson@hhs.se

Page intentionally left blank

Contents

I. Introduction **1**

II. Literature: Machine Learning in Asset Pricing **6**

III. Data and Empirical Framework **9**

III.A Data Sources 9

A1. Firm-Characteristics 10

A2. Bond Characteristics 11

III.B Empirical Estimation Framework 11

B1. Generative Adversarial Network 13

B2. Feedforward Neural Network 16

B3. Elastic Net 16

B4. Permanent-Transitory Decomposition and GAN 17

B5. Risk Loadings 23

B6. Evaluation Metrics 24

B7. Model Architecture and Hyperparameter Selection 26

IV. Empirical Results for Nordic Equities **27**

IV.A Asset Pricing Performance 27

IV.B Predictive Performance 31

IV.C Variable Importance 35

IV.D Pricing Characteristic Portfolios 39

IV.E Interactions and Nonlinearities 42

V. Discussion **46**

VI. Concluding Remarks **53**

References **55**

Appendix **62**

A1. Supplementary Results and Hidden States 62

A2. Additional Characteristic Pricing Tables 70

A3. Additional Characteristic Interactions 75

A4. Additional Model Descriptions 80

A5. Supplementary Data Description 88

A6. Result of Validation Procedure 98

A7. Fama-French Robustness Checks 113

I. Introduction

Using a novel weekly dataset of Nordic stocks, I make two contributions to the literature. First, I compare the asset pricing performance of three machine learning models; a generative adversarial network (GAN), a feedforward neural network (FFN), and an elastic net (EN) on this novel dataset. The GAN model outperforms the other models in (i) separating the signal from the noise, (ii) producing a more robust expected-return and risk-loading relationship, and (iii) achieving a higher explanatory in individual stock returns and in the cross section of expected returns. Specifically, the baseline GAN model generates a tradable portfolio that achieves an annualized Sharpe ratio of 2.41 on out-of-sample data. This is compared to 1.90 for the EN, 1.29 for the FFN, and 0.72 for the market-weighted portfolio of all stocks. This paper is the first to perform a side-by-side evaluation of these established machine learning models on international stock data and uses higher frequency, lower signal-to-noise data than in the previous literature. By building risk loadings on the model-implied mean-variance optimal portfolios, market participants can find a strong linear relationship to future expected returns and price Nordic stocks accordingly. The second contribution is the introduction of an extended GAN model that is motivated by the permanent-transitory decomposition of the pricing kernel. Using two different metrics of explained variation, I find that imposing this economic constraint improves the ability of the GAN model to explain individual returns and their means. The baseline GAN model explains 10% of individual stock returns and 63% of average returns on out-of-sample data. These numbers improve to 14 - 15% and 65 - 66% for the extended GAN models. Although these incremental gains in explanatory power are minor in their economic magnitude, they are ubiquitous and robust to different test assets. These gains persist when evaluated on 460 characteristically managed portfolios. Since all extended models share the same level-shift improvement in explanatory power, the gain is directly attributable to the permanent-transitory decomposition. By imposing this new economic constraint in 3 different ways, I also find that the signal-to-noise ratio can be improved from 2.41 to 2.85, but only when the constraint is applied in an economically informed way. When the constraint is applied naively, the ability of the model to separate the signal from the noise deteriorates slightly to 2.12. By ranking the variable importance of all firm characteristics, I conclude that characteristics related to market frictions and other limits-to-arbitrage (LTA) are the main drivers of the machine learning models. This finding contributes to an expanding body of literature documenting that machine learning models often confuse LTA-related predictable patterns for economically interesting systematic risk. By visualizing pairwise interactions, it is also apparent that the best performing machine learning models capture interactions between predictors, but are otherwise approximately

linear in predictors. This suggests that it is more important to properly capture variable interactions than other, potentially more complex nonlinear patterns.

The GAN, FFN, and EN have received prominence in the literature due to their strong link to the underlying asset pricing theory. All asset pricing models *must* produce a pricing kernel (PK), also called a stochastic discount factor (SDF) (Cochrane, 2005). The pricing kernel expresses the systematic behavior of all assets in the financial markets. Therefore, researchers and market participants must build good models on the pricing kernel to be able to make policy recommendations or real-world investment decisions. At any point in time, the pricing kernel fully condenses the overall macroeconomic state into a single number. Crucially, the risk-reward relationship of all financial assets also depends on this pricing kernel. The covariance of any asset to the pricing kernel reflects the riskiness of that asset, which directly translates into the expected return premium of the asset. The consensus for the market price of any asset comes from discounting its future expected payoffs by the risk premium implied by the pricing kernel. Crucially, these expectations are based on the aggregated information set available to investors. Therefore, to price any asset today, an asset pricing model must make predictions about the future, implicitly or explicitly.

The large size of this information set pushes the field of asset pricing towards a big data regime. Significant technological developments, such as high-capacity computing and widespread data availability, have enabled researchers to study asset prices using substantially larger datasets. Although a modern-day econometrician has access to thousands of predictors at a high frequency, not all of them are informative. For some predictors, informational content can only be extracted after the data have been properly cleaned and/or transformed. Furthermore, the asset pricing modeler has to make an economically motivated choice about the functional mapping from the set of predictors onto the pricing kernel. All of these choices are subject to the experience and bias of the researcher. In general, researchers have addressed these issues by imposing a high degree of sparsity. That is, by selecting a few risk factors and specifying the pricing kernel as a linear factor model (Nagel, 2021). These low-dimensional linear factor structures have seen significant success empirically; the most prominent example is the market, size, and value factors of Fama and French (1992). However, researchers have expanded the list of informative predictors over the years. Currently, the 'factor zoo' contains hundreds, if not thousands, of factors considered informative in asset pricing (Cochrane, 2011). If there are indeed that many informative factors, then relying on a sparse and linear representation of the pricing kernel is unlikely to advance the field of asset pricing further.

These two challenges; (1) the large information set and (2) the ambiguous functional mapping from predictors to the pricing kernel, represent a typical setting in which machine learning models tend to outperform standard econometric tools. The distinction is subtle,

but machine learning differs from the standard econometric approach in several ways. The reason why machine learning is particularly adept at big data problems is that feature extraction is performed explicitly and implicitly in a data-driven way by transformations of the models. (Athey, 2019). In contrast, in the standard econometric framework, the econometrician has to manually perform feature engineering. In terms of objectives, machine learning is geared towards prediction and classification, not hypothesis testing and causality (Kleinberg et al., 2015). Hypothesis tests and confidence intervals on parameters of deep learning models are generally not possible, except in a few restrictive cases. In fact, the estimated parameters are usually ignored in favor of the model predictions. The flexible model architecture of machine learning models allow them to capture nonlinear relationships, whereas econometric methods have limited capacity to capture nonlinear relationships that are not already specified by the researcher. This is why machine learning methods are better suited to capture the ambiguous relationship from predictors to the pricing kernel. This flexibility reduces the bias of the model, but it may come at the expense of overfitting, which could lead to a high variance in performance across samples. This is why much of the machine learning literature is centered around the notion of the bias-variance trade-off. The training of machine learning models is usually stochastic, performed in mini-batches, and validated using out-of-sample prediction-related metrics, whereas econometrics focuses on standard goodness-of-fit measures. Econometric models are usually carefully specified in order to find the one "true" model that includes only the most informative predictors, whereas the machine learning literature embraces weak learners if they can be combined into ensembles (Varian, 2014). Machine learning models are often criticized for lacking transparency and offering little or no interpretable economic information compared to standard econometric models (e.g. Ribeiro et al., 2016). This "black box problem" can be mitigated with tools such as input heatmapping, feature detection visualization, and diagnostic classification (Zednik, 2021). I adopt some of these tools in this paper as it allows researchers to extract additional economic content and build more explainable models.

The problem of finding a function that maps from a set of predictors to the space of asset prices is a supervised regression problem (Hull, 2021). While there are readily available machine learning methods that have been adapted to these types of problems; the econometrician must be cautious to apply the methods "off-the-shelf" in an asset pricing setting. There are a few key domain-specific features in asset pricing that differentiate it from a traditional machine learning setting. Most importantly, asset pricing has a measurement problem, the signal-to-noise ratio is very low compared to traditional machine learning problems (Nagel, 2021). Asset pricing theory makes predictions on the *expected returns* of the assets. However, this is only a small part of the returns observable to the researcher in the data. Because

expected returns cannot be observed, the econometrician will inevitably have to separate this expected return (the "signal") from the idiosyncratic noise present in the data. If the researcher has done a good job, then the pricing kernel produces a high signal-to-noise ratio. Although the availability of high-frequency data has increased dramatically, there is little to no gain to be had in statistical power for these estimates of the first moment (Merton, 1980). In the absence of a theoretical model of expected returns, deriving reliable expected return estimates requires many decades of data, and most available datasets go back only a few decades. Additionally, aggregated results (i.e., portfolio outcomes) are typically of interest in asset pricing, whereas machine learning typically focuses on individual examples. The covariance of prediction errors is also an important risk factor in asset pricing, but is largely treated as a statistical nuisance and ignored in traditional machine learning (Nagel, 2021). Although there have been important advances in big data handling, the data available for asset pricing studies may only be a small subset of the overall information set available to investors, which is the traditional Hansen and Richard (1987) critique. Markets are also fundamentally adaptive (e.g. Lo, 2019) which means that investors can learn about the underlying state of the market over time and are quick to adapt to structural breaks, such as technological advances and changes in regulatory regimes. These structural breaks could imply that models that are trained under one regime may break down when evaluated out-of-sample in the new regime. To conclude, the "kitchen sink" approach to machine learning in asset pricing is unlikely to perform as well as machine learning models with economically motivated model architectures, objective functions, and validation procedures (Avramov et al., 2023).

Research Questions: GAN, FFN, and EN

7 research questions (RQ) will guide the comparative analysis of the GAN model of Chen et al. (2023a), the FFN model of Gu et al. (2020), and the EN model of Kozak et al. (2020):

RQ 1: Which model finds the strongest signal out-of-sample?

RQ 2: Do all models outperform the benchmark CAPM model in terms of Sharpe ratio?

RQ 3: Do the models hold any (i) predictive power and (ii) explanatory power?

RQ 4: Are the machine learning models already priced by the benchmark CAPM model?

RQ 5: How important are frictions in driving the results of the models?

RQ 6: What are the most important characteristics for pricing Nordic stocks?

RQ 7: Do interactions and nonlinearities in characteristics matter for pricing Nordic stocks?

The answers to RQ 1 - RQ 4 constitute one of two main results of the paper. The above research questions are motivated by the apparent bias in the literature to evaluate machine learning models on US stock data at the monthly frequency, almost exclusively. This bias is due to the convenience of having readily available and pre-cleaned time series from the CRSP database. Although Chen et al. (2023a) finds that the GAN can produce a higher signal-to-noise ratio than reported for other models, the question is whether this result is specific to US stocks, or only holds at the monthly frequency. This paper is the first to do a side-by-side evaluation of the GAN, FFN, and EN models using international stock data and the weekly frequency. This makes it possible to pass judgment whether the performance of whether previous results are robust to changes in the empirical setting. Critically, international stocks typically follow different dynamics from US stocks.[1] Furthermore, return dynamics shift with data frequency. Because monthly log returns are just an aggregation of weekly returns, the Central Limit Theorem (CLT) says that weekly returns should to operate in a lower signal-to-noise, possibly more asymmetric, and necessarily more leptokurtic environment than monthly returns (Campbell et al., 1997).[2] As machine learning models generally perform better in higher signal-to-noise environments (Nagel, 2021), the question is whether the performance deteriorate in the new lower signal-to-noise ratio setting.

The answers to RQ 5 - RQ 7 give an economic interpretation of the main result. RQ 5 and RQ 6 contribute to an ongoing debate on whether machine learning models generate high signal-to-noise ratios by loading on pseudo-predictable patterns that are related to market frictions, illiquidity, and other limits-to-arbitrage (LTA) as opposed to economically interesting signals (see Avramov et al., 2023). RQ 7 contributes to the debate on nonlinearities in asset pricing models. Some find that machine learning models perform well because of their ability to capture very complex nonlinearities (see, e.g. Gu et al., 2020 and Bryzgalova et al., 2023) while others suggest that the models create economic gains by capturing interactions instead of other complex nonlinear patterns (e.g. Chen et al., 2023a).

Research Questions: Permanent-Transitory Decomposition

The following research questions are related to the extended GAN models that consider the permanent-transitory decomposition in the specification of the pricing kernel:

RQ 8: Do the extended GAN model provide any economic gains in terms of (i) signal-to-noise

[1] Some factors that affect the behavior of international stocks include; country-specific risk premia (see for example Erb et al., 1995), international market segmentation (e.g. Camanho et al., 2022), and heterogeneity in financial development (e.g. Maggiori, 2017).

[2] In general, the signal-to-noise ratio *increases* by a factor of $\sqrt{k}$ as the frequency of data is *decreases* by a factor k (Campbell et al., 1997).

ratio or (ii) explanatory power?

RQ 9: Are both components of the pricing kernel (i) driven by frictions, and (ii) are they highly nonlinear in predictors?

Here, RQ 8 is the second main contribution of the paper and RQ 9 provides a supplementary economic interpretation of this result. These questions are motivated by the contemporary school of thought that economically informed constraints can improve the performance of machine learning in an asset pricing setting and that uninformed use of machine learning models does not perform as well on out-of-sample data. (see, e.g., Nagel, 2021 and Avramov et al., 2023). By imposing the permanent-transitory decomposition on the GAN model, this paper contributes by putting the notion "machine learning + economic constraints = more informed asset pricing model" to the test.

II. Literature: Machine Learning in Asset Pricing

This paper uses pre-existing machine learning models developed separately in three seminal papers in the literature. The first is Chen et al. (2023a), who introduces the generative adversarial network as a conditional asset pricing model. The authors use the long short-term memory (LSTM) model of Hochreiter and Schmidhuber (1997) to estimate hidden macroeconomic states and adopt a variant of the GAN iterator of Goodfellow et al. (2014). The authors then build a robust pricing kernel by solving the adversarial generalized method of moments (GMM) problem of Hansen and Jagannathan (1997). This no-arbitrage constrained pricing kernel achieves an unprecedented out-of-sample Sharpe ratio and explanatory power on monthly US stock data. The second paper is Gu et al. (2020), which compares the performance of feedforward networks with standard machine learning tools such as regression trees and elastic nets. They find that the FFN model is better able to solve typical return prediction problems because of its added flexibility. Using model-implied expected returns, the authors are able to produce a high signal-to-noise pricing kernel using monthly US stock data. Finally, in their landmark study, Kozak et al. (2020) uses the dual-penalty elastic net of Zou and Hastie (2005) to minimize an objective function based on the HJ distance of Hansen and Jagannathan (1991). As a result, the authors are able to identify a sparse linear representation of the empirical pricing kernel on monthly US stock data that has a minimal model specification error in-sample. They also find that the regularization imposed by the penalties directly translates to prior beliefs about the maximum attainable signal-to-noise ratio. The pricing kernel tends to perform the best when combined with economically motivated priors, that is, when the prior is in line with historical estimates of the Sharpe ratio.

Avramov et al. (2023) survey the state of the literature on machine learning in asset pricing and identify persistent themes across the field. Regardless of machine learning model, they tend to improve in asset pricing tasks when combined with economically motivated constraints. These economic constraints can be implemented in a multitude of ways. Examples include carefully designed objective functions, adapted model architectures, or regularization techniques adapted to the asset pricing setting. Freyberger et al. (2020) adopts the Lasso model of Tibshirani (1996) to select informative predictors. Using an economically motivated penalization, the model performs better on monthly US stock returns than unpenalized linear models. Rossi (2018) use boosted regression trees (BRTs) to predict monthly returns of US stocks, finding that trees can outperform simple linear models. Similarly, Moritz and Zimmermann (2016) uses tree models to classify monthly US stocks into portfolios that achieve an information ratio twice as high as the linear Fama-MacBeth regression benchmark. Using monthly returns of US stocks, Bryzgalova et al. (2023) adds economic constraints in the form of mean and covariance shrinkage to regression trees. This asset pricing tree model (AP-Tree) includes a pruning technique that combines a gradient-boosted sequential partition of predictors with the no-arbitrage constraint. The AP-Tree model uses ensemble learning to combine many weak learners into an aggregated pricing kernel. This procedure leads to economically meaningful gains in the signal-to-noise ratio.

Bianchi et al. (2021) uses the FFN model of Gu et al. (2020) on bond data and finds that the model also performs well in prediction problems for monthly US bond returns. Messmer (2017) extends these two studies by considering deeper feedforward networks (DFN) on US monthly stock returns, finding that the turnover is very high. Only by imposing restrictions on portfolio rebalancing do the models produce positive returns adjusted for expected trading costs. Cong et al. (2021) compare different type of recurrent neural networks (RNNs) and find that the LSTM model has the best asset pricing performance out-of-sample. Nagel and Singleton (2011) uses a constrained version of the adversarial objective function as in Chen et al. (2023a). The authors impose a sparse representation of the pricing kernel, conditional on a few aggregated predictors. They use this technique to build optimal characteristically managed portfolios using monthly US stock data. Cong et al. (2020) introduces a framework for portfolio allocation rules that vary in time and state using reinforcement learning (RL). By adjusting for various trading frictions they find that the best economic constraint is to use a risk-reward metric, such as the Sharpe ratio, when specifying the objective function.

In terms of unsupervised learning, Kelly et al. (2019) applies instrumented principal component analysis (IPCA) to build a conditional asset pricing model using monthly US stocks. Similarly to standard principal component analysis (PCA), the authors use IPCA to perform a data-driven linear dimensionality reduction. Gu et al. (2021) extend this idea by

introducing the variational auto-encoder (VAE) originally developed by Kingma and Welling (2013) to asset pricing. The authors combine the VAE model with the no-arbitrage constraint to reduce the dimension of the predictors nonlinearly. Using monthly US stock returns, the authors show that the VAE model is able to reduce pricing errors compared to conventional sparse linear factor models. Heaton et al. (2017) uses a VAE model for hierarchical decision problems, finding that VAEs can detect robust patterns in predictor interactions. However, the patterns lack an economic explanation in the underlying asset pricing theory.

Literature: Permanent-Transitory Decomposition

Alvarez and Jermann (2005) were first to show sufficient conditions to decompose the pricing kernel into a permanent and transitory component. If the pricing kernel and asset returns are jointly stationary and ergodic, then a non-trivial permanent martingale component must exist. Additionally, a non-trivial transitory component representing the yield on an infinite-duration zero-coupon bond (ZCB) must also exist. They construct variance bounds for each component in the same vein as Hansen and Jagannathan (1991) did for the overall pricing kernel. Empirically, the authors find that the permanent component tend to be more volatile. In a Markovian environment, Hansen and Scheinkman (2009) show that these components are closely related to a Perron-Frobenius eigenfunction problem. In particular, the eigenvalue represents the yield on long-term state-contingent claims and the eigenfunction maps from the Markov states onto the prices of such claims.

Borovi̇čka et al. (2011) apply the decomposition for structural models in order to derive implied shock-price elasticities. These measures are used to decompose the risk premium resulting from different types of undiversifiable macroeconomic shocks. Hansen and Scheinkman (2012) established sufficient conditions for the coexistence of a decomposed pricing kernel with the recursive utilities of Epstein and Zin (1989). Hansen (2012) uses this result to build an empirical decomposed pricing kernel with recursive preferences. Bakshi and Ghabi-Yo (2012) extends the eigenfunction theory of Hansen and Scheinkman (2009) in order to produce stronger variance bounds for the components compared to the bounds of Alvarez and Jermann (2005). They also construct variance bounds on the ratio of the component variances. The authors also find empirical evidence to suggest the components have a positive covariance. This empirical result is contrary to the implications of the long-term risk model of Bansal and Yaron (2004). Backus et al. (2014) found that the permanent component has the largest variance and that most models on preferences cannot reproduce the high permanent variance without also predicting unrealistic term spreads.

Christensen (2017) extend the Perron-Frobenius theory of Hansen and Scheinkman (2009)

to a nonlinear Perron-Frobenius problem and show how to solve it nonparametrically. Qin and Linetsky (2016) extent the theory to a much less restrictive setting where the pricing kernel is a positive semimartingale that depends on continuous-time Markov state processes. Crucially, they established sufficient conditions for the uniqueness and existence of the permanent-transitory decomposition in this setting. This result allowed Qin and Linetsky (2017) to show that the decomposition must hold if the pricing kernel is a semimartingale in a no-arbitrage environment. This is a much less restrictive assumption than that of Alvarez and Jermann (2005), suggesting that the permanent-transitory decomposition is a feature rather than a bug in the pricing kernel.

III. Data and Empirical Framework

III.A Data Sources

I use firm data and market data on small, medium, and large-cap stocks listed on the Nordic market division of the Nasdaq Nordic exchange. All data are on a weekly frequency and were retrieved from Refinitiv Eikon. All listed preferred stock instruments are excluded and the class B share is used as the representative instrument.[3] The Oslo Børse is excluded as it was never part of Nasdaq Nordic, and Nasdaq First North is also excluded as it contains only the most illiquid stocks. The Nasdaq Nordic was first introduced in 2003 but the study period is extended to the $T = 1251$ weeks spanning 2000/01/04 to 2023/12/19. The dataset is filled backward to the beginning of 2000 by including stocks that were listed on the country-specific stock exchanges in Sweden, Denmark, Finland, and Iceland at that point in time. Stocks enter and exit the sample as they are listed and delisted, which eliminates any effects of survivorship bias. Two additional filters are applied. First, stocks leave the sample the following week if their total return index since the IPO falls below 1%. Second, stocks that lose more than 90% of their value in one week exit the sample the following week. These filters are applied to identify "ghost" firms that still satisfy the Nasdaq listing criteria but are effectively out of business. Firms that exit the sample can reenter at a later date by relisting on Nasdaq Nordic. The final dataset is an unbalanced panel containing $N = 810$ Nordic stocks; see Figure A16 for the number of stocks in the sample in any week.

I work with excess stock returns $R^e_{t+1,i} = R_{t+1,i} - R^f_{t+1}$ for firm i in week t. The gross returns $R_{t+1,i}$ are adjusted for all corporate actions and reinvested dividends. The risk-free rate R^f_{t+1} used is the weekly yield on the shortest-duration government T-bill in the country

[3] If unavailable, the share class is selected by order of the alphabet

of origin of the stock exchange.[4] After all filters have been applied, the weekly excess returns are winsorized at ±90% as a post-processing step. The study period is partitioned into 3 subsamples; (i) the training period 2000/01/04 to 2011/12/20, (ii) the validation period 2011/12/27 to 2015/12/22, and (iii) the test period 2015/12/29 to 2023/12/19. This means that there are $T_{\text{train}} = 625$, $T_{\text{valid}} = 209$, and $T_{\text{test}} = 409$ weeks of training, validation, and test data, respectively. Models are trained on the training dataset, hyperparameters are validated on the validation dataset, and out-of-sample performance is tested on the test dataset.

Firm Characteristics

I construct $K^{\text{char}} = 46$ firm characteristics for each stock. The characteristics are derived from firm-specific accounting or market data and are grouped into 6 categories: value, profitability, investment, frictions, intangible factors, and returns; see Table A7 for details of the construction of each characteristic. All data denominated in local currencies are converted to euros before the characteristics are constructed. The characteristics used in this study are the same as in Chen et al. (2023a), but adjusted where necessary to fit the Nordic and weekly setting. Variables from the income or cash flow statement are adjusted to trailing-twelve months (TTM) measures where possible. Balance sheet variables are updated to the latest available metric from interim or annual reports. Because this paper operates in a higher frequency environment than comparable studies, including TTM measures and interim updates of accounting data is an important extension. Additionally, the higher frequency of data makes the time series dimension larger in relation to the cross-sectional dimension when compared to the panel structure used in earlier studies.[5] The characteristics are lagged by 9 weeks (c.a. 2 months) to eliminate the risk of inadvertent information slippage. This forces the models to identify persistent signals rather than building swing trading strategies around reporting dates. All characteristics $\tilde{C}_{t,i}$ are cross-sectionally ranked between all N_t firms in the sample in week t and subsequently quantile-normalized (QN):

$$C_{i,t} = \left(\frac{\text{rank}(\tilde{C}_{i,t}) - \frac{1}{2}}{N_t + 1} \right) - \frac{1}{2} \tag{1}$$

This transformation maps all characteristics to the range $[-0.5, 0.5]$ where the median (and mean) of the firm characteristics are 0. This transformation is conventional in the literature (see, e.g., Nagel, 2021 and Freyberger et al. and 2020). Median imputation is used, which is not used by Chen et al. (2023a) but is otherwise common practice in the

[4] e.g. firms listed on the Stockholm Stock Exchange use the 1-month Swedish T-bill as a riskless proxy.

[5] The panel used in this study has a time series dimension of $T = 1,251$ weeks and a cross-section of $N = 810$ firms. In comparison, Chen et al. (2023a) has $T = 588$ months and $N = 10,000$ firms

literature (see, e.g., Gu et al., 2020 and Kozak et al., 2020). These 46 variables constitute the information set of firm characteristics $I_{i,t} \in \mathbb{R}^{N \times K^{\text{char}}}$ available in week t for firm i. Ignoring imputed firms, the 46 unnormalized firm characteristics $\tilde{C}_{t,i}$ are also aggregated every week by their cross-sectional median. Standard transformations are performed on these $K^{\text{macro}} = 46$ macroeconomic time series to remove apparent nonstationarities; see Table A8 for details of this procedure.[6] Like the firm characteristics, these aggregated variables are lagged by 9 weeks and constitute the macroeconomic information set $I_t \in \mathbb{R}^{K^{\text{macro}}}$ available at time t.

Bond Characteristics

The bond data used in this study is $N^{(\text{bond})} = 165$ aggregate bond indices for Sweden, Denmark, Finland, and Iceland; see Table A10 for details of the indices used. These are indices of government bonds, corporate bonds, mortgage bonds, or a mix thereof.[7] All data is retrieved from Refinitiv Eikon and the bond data is at the weekly frequency to synchronize with the frequency of the firm data. Data originally denoted in local currencies are converted to euro and $K^{\text{bond}} = 30$ bond characteristics are created for each bond index; see Table A9 for details on the construction of each bond characteristic. Like firm characteristics, bond characteristics are ranked weekly and normalized by quantile to the range $[-0.5, 0.5]$ according to Equation (1). Missing bond characteristics are imputed using the median, i.e. set to zero, and all bond characteristics are lagged by 9 weeks. For bond b in week t, these 30 bond characteristics constitute the information set for the bond index $I^{(\text{b})}_{b,t}$. Macroeconomic variables are constructed from these bond characteristics by aggregating them at the cross-sectional median every week t. The most apparent nonstationarities are removed using the transformations defined in Table A11. Like firm characteristics, the set of macroeconomic variables $I^{(\text{b})}_t$ for bonds is lagged by 9 weeks. The excess bond returns $R^{e,(b)}_{t+1}$ are calculated by taking the weekly gross return of the aggregate bond index $R^{(b)}_{t+1}$ in excess of the risk-free rate R^f_{t+1} in the local currency in which the index is denoted.

III.B Empirical Estimation Framework

The empirical setting used in this paper is set within the no-arbitrage framework. Specifically, the excess return $R^e_{t+1,i}$ of stock i must satisfy the fundamental conditional pricing equation:

[6] Unlike many econometric models, most machine learning models do not require stationary features. As noted in Chen et al. (2023a); If stationarity is required, LSTMs and FFNs are more than flexible enough to perform these feature extractions implicitly.

[7] Bond indices are used due to the limited data available on Nordic company-specific corporate bonds dating back to the turn of the millenium

$$\mathbb{E}_t \left[M_{t+1} R^e_{t+1,i} \right] = 0 \tag{2}$$

Where a strictly positive pricing kernel $M_{t+1} = M(\mathcal{I}_t)$ is a necessary and sufficient condition to place us in the no-arbitrage setting (Campbell, 2018).[8] $\mathbb{E}_t[\cdot]$ denotes the expectation operator conditional on the time-t information set $\mathcal{I}_t$. In the empirical estimation, this information set contains both past and present firm-specific and macroeconomic information $\mathcal{I}_t = [I_t, I_{t,i}]$. Because I do not assume markets are complete, the pricing kernel is not necessarily unique. Instead, one pricing kernel has to be selected from a larger set of admissible candidate kernels.[9] Following Chen et al. (2023a) and Kozak et al. (2020), I project the pricing kernel onto the space of returns without losing generality:

$$M_{t+1} = 1 - \omega_t' R^{(\text{basis})}_{t+1} = 1 - F_{t+1} \tag{3}$$

where $R^{(\text{basis})}_{t+1}$ denotes a vector of returns on basis assets. The basis assets are used to build the pricing kernel. In the empirical estimation, the basis assets are the set of excess stock returns available in the investment opportunity set; $R^{(\text{basis})}_{t+1} = R^e_{t+1}$. At each time t, the vector $\omega_t \in \mathbb{R}^{N^{(\text{basis})}}$ is the tradable portfolio of the basis assets. The affine transformation in Equation (3) corresponds to the pricing kernel that lies on the minimum variance bound of Hansen and Jagannathan (1991). As a result, the tradable return factor F_{t+1} will be conditionally mean-variance efficient. Solving for the pricing kernel in the fundamental moment equation (2) is a conditional generalized method of moment (GMM) problem as defined in Hansen (1982). In the no-arbitrage setting, the conditional moment equation implies an unconditional moment equation with infinite moment conditions (Chen et al., 2023b):

$$\mathbb{E} \left[M_{t+1} R_{t+1} Z_t \right] = 0 \tag{4}$$

Where $Z_t = Z(\mathcal{I}_t)$ denotes $\mathcal{F}$-measurable instruments. In the empirical analysis, the instruments are treated as a vector-valued function $Z : \mathbb{R}^{K^{\text{char}}} \times \mathbb{R}^{K^{\text{macro}}} \to \mathbb{R}^D$. The D instruments convey conditioning information and are used to construct $N^{(\text{basis})} \cdot D = N^{(\text{test})}$ characteristically managed portfolios $R^{(\text{test})}_{t+1,j} = \sum_{i=1}^{N} R^e_{t+1,i} Z_{t,j}$. These test assets $R^{(\text{test})}_{t+1}$ are used to test the performance of the admissible pricing kernels, allowing us to identify the best one.[10] When the basis assets used to build the pricing kernel are the same as the test

[8] Other names for the pricing kernel includes stochastic discount factor, state-price density, marginal rate of substitution (MRS), and change-of-measure (Cochrane, 2005).

[9] M-uniqueness and other properties of complete markets are explored in Dybvig and Ingersoll Jr. (1982)

[10] The test assets $R^{(\text{test})}_{t+1}$ are sometimes called synthetic payoff vectors (see e.g., Hansen and Sargent, 2001)

assets used to test its performance, i.e. when $R_{t+1}^{(\text{basis})} = R_{t+1}^{(\text{test})}$, then the GMM problem is exactly identified with a closed-form solution (Chen et al., 2023b):

$$\omega_t^* = \mathbb{E}\left[R_{t+1}^{(\text{basis})} R_{t+1}^{(\text{basis})\prime}\right]^{-1} \mathbb{E}\left[R_{t+1}^{(\text{basis})}\right] \tag{5}$$

here ω_t^* is the conditionally mean-variance optimal portfolio. The empirical setting used in this study is overidentified, that is, $N^{(\text{test})} > N^{(\text{basis})}$. In such a setting, Hansen and Jagannathan (1997) showed that the pricing kernel that solves the adversarial GMM problem below is closest to the true pricing kernel in the least-squares sense:[11]

$$\{\omega^*, Z^*\} \in \arg\min_{\omega(\mathcal{I}_t)} \max_{Z(\mathcal{I}_t)} \left\| \mathbb{E}\left[\left(1 - \omega_t' R_{t+1}^{(\text{basis})}\right) R_{t+1}^{(\text{test})}\right] \right\|_2^2 \tag{6}$$

the inner maximization problem chooses the optimal instrument function Z^* in a data-driven way to maximize pricing errors. In contrast, the outer minimization procedure selects the optimal portfolio function ω^* that minimizes the worst possible pricing error.[12]

Generative Adversarial Network

Chen et al. (2023a) introduce the generative adversarial network (GAN) to no-arbitrage asset pricing by constructing an objective function $\mathcal{L}$ based on the empirical version of the optimization problem in Equation (6):

$$\mathcal{L}(\omega, Z \mid \mathcal{I}_t) = \frac{1}{N} \sum_{i=1}^{N} \frac{T_i}{T} \left\| \frac{1}{T_i} \sum_{t=1}^{T_i} \left(1 - \omega_t' R_{t+1,i}^e\right) R_{t+1,i}^e Z_t \right\|^2 \tag{7}$$

where the moment conditions can be interpreted as squared pricing errors, which are weighted by the convergence rates $\sqrt{T_i}/T$. This allows a firm i with a longer time series T_i to contribute more to the objective than a firm with fewer observations. Chen et al. (2023a) estimate the functions ω_t and Z_t in an iterative zero-sum game using two networks; (i) the Pricing Kernel (PK) network and (ii) the conditional (COND) network. Each network consists of a long short-term memory (LSTM) cell and a feedforward network; see Figure I for the architecture of the GAN model. The estimates of ω and Z are conditional on the most recent information set $\mathcal{I}_t$ and the current state of the opposing network:

[11] It also minimizes the model specification error (also called the Hansen-Jagannathan (HJ) distance).

[12] Hansen and Jagannathan (1997) suggests that the adversarial nature of this estimation makes the final estimates more robust compared to regularized GMM. It is also less likely to regularize away weak but informative predictors.

$$\hat{\omega} \in \arg\min_{\omega} \mathcal{L}(\omega|\hat{Z}, \mathcal{I}_t) \quad \text{and} \quad \hat{Z} \in \arg\max_{Z} \mathcal{L}\left(Z|\hat{\omega}, \mathcal{I}_t\right) \tag{8}$$

The purpose of the LSTM is to build a function $\hat{h}_t = h(I_t)$ that extracts k hidden states from the set of macroeconomic variables I_t. To be specific, the LSTM learns the function $h : \mathbb{R}^{K^{\text{macro}}} \to \mathbb{R}^k$, where the optimal number of states k is selected using a data-driven validation procedure.[13] An LSTM is particularly adept at capturing long-term dynamics in time series and can take into account many past lags of all predictors. This LSTM cell is necessary because it is difficult to assess the current state of the business cycle by only conditioning on the latest observation of a long sequence of macroeconomic data (Chen et al., 2023a). For a detailed description of LSTM cells; see Appendix A4. Using the estimated hidden states h_t, the FFN model learns a function that maps from the hidden state and the firm characteristics $I_{t,i}$ to individual portfolio weights $\hat{\omega}_{t,i}$ within the pricing kernel. In detail, the function $\omega_{t,i} = \omega(\hat{h}_t, I_{t,i})$ is learned. In a post-processing step, the weights are l_1-normalized weekly; $\omega_t / \|\omega_t\|_1$. This preserves the original sign and ranking of the portfolio weights, but scales the overall magnitude of the positions to economically meaningful levels. Similarly, for the COND network, an FFN model learns a function to transform hidden states $\hat{h}_t^Z$ and firm characteristics into instruments; that is, $\hat{Z}_t = Z(\hat{h}_t^Z, I_{t,i})$. For a detailed overview of a FFN; see Appendix A4.

The empirical estimation procedure is detailed in Algorithm (1).[14,15] The parameters θ of the PK network and φ of the COND network include the weights and biases of the four gates of the LSTM cell and parameters of the FFN model. The training process is carried out in two phases. It begins with an unconditional phase where the test assets are fixed to the basis assets; $R_{t+1}^{(\text{test})} = R_{t+1}^{(\text{basis})}$ and the PK network minimizes $\mathcal{L}$ without considering the adversary. Then comes a conditional estimation phase. In particular, the COND network chooses the optimal instruments Z_t and arbitrage strategies $R_{t+1}^{(\text{test})}$ to maximize $\mathcal{L}$. In each iteration, the PK network responds by strengthening the pricing kernel to minimize $\mathcal{L}$ conditional on these test assets. In this iterative process, the parameters $\theta^{(\text{PK})}$ and $\varphi^{(\text{COND})}$ of each network are updated conditional on the parameters of the opposing network.[16] This process continues until a stable evolutionary equilibrium is reached, that is, where neither model can make significant improvements against the other (Hull, 2021).[17]

Selecting the D most informative test assets to proxy for the infinite moment conditions

[13] The hidden states are normalized to $[-1, 1]$ by batch in the empirical estimation.

[14] This is the AdversarialGMM algorithm of Lewis and Syrgkanis (2018) adapted to the asset pricing setting.

[15] Model architectures, performance metrics, and loss functions are custom-built using Keras for Python

[16] The parameters of the opposing network are always frozen during the training process.

[17] Alternative outcomes include mode collapse, where a network repeatedly generate the same object. If there is a capacity imbalance in the networks, one could simply dominate the other (Goodfellow et al., 2014)

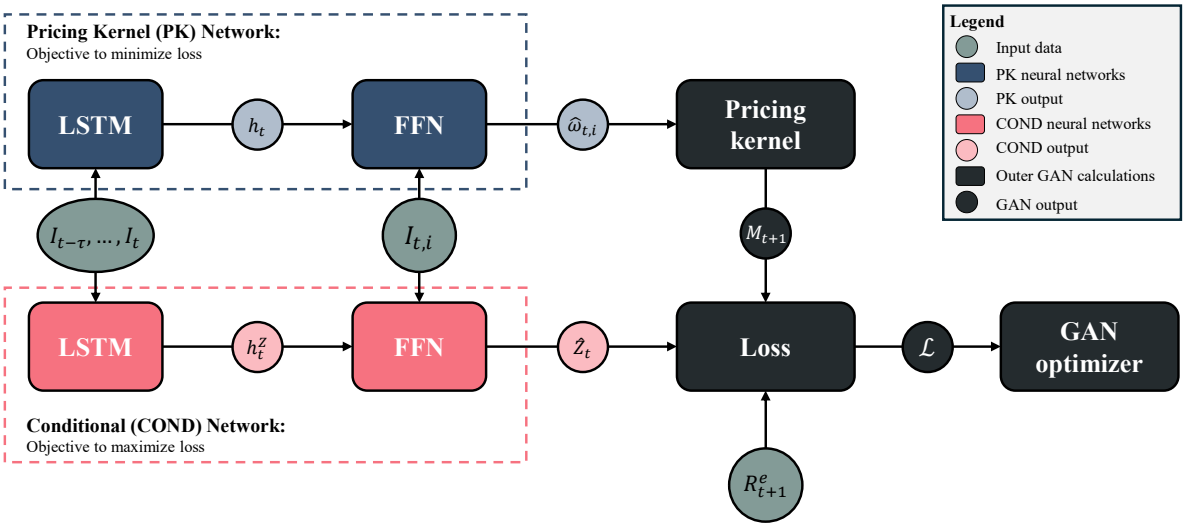


Figure I
GAN Model Architecture

Notes. This figure illustrates the architecture of the generative adversarial network (GAN) model of Chen et al. (2023a). The figure is adapted from Figure (1) in the February 2023 version of the original authors' paper. The GAN consists of the PK Network (blue), the COND Network (pink), and the outer GAN iterator (black). Each network consists of a long short-term memory (LSTM) cell and a feedforward network (FFN). The LSTM transforms macroeconomic information I_t (with lookback τ) into hidden states h_t and h_t^Z. FFNs transform hidden states and firm characteristics $I_{t,i}$ to portfolio weights $\hat{\omega}_{t,i}$ and a vector of instruments $\hat{Z}_t$. These are combined with stock returns R_{t+1}^e to produce a pricing kernel M_{t+1} and a loss function $\mathcal{L}$.

is a challenge. Econometricians typically approach this problem by choosing common test assets based on decile or double-sorted portfolios based on characteristics such as size and value (e.g. Nagel and Singleton, 2011). The problem is that it is difficult to know ex ante whether the preselected test assets are appropriate. These selections are also subject to researcher experience and bias. This is a major issue because all empirical results become preordained from this point on. By letting a neural network find the optimal instruments Z in a data-driven way, Chen et al. (2023a) were the first to merge the notion of robust control in the sense of Hansen and Sargent (2001) with semiparametric and nonparametric estimation in asset pricing (Kelly and Xiu, 2023). To be specific, the adversary provides robust control by iteratively pointing out all faults in the pricing kernel in a data-driven way. Intuitively, the adversary favors the trading signals that the pricing kernel is unable to price well. In this way, it behaves similarly to an arbitrageur that exploits mispricings in the market. Therefore, the economic interpretation of Algorithm (1) is as a zero-sum game between an asset pricing modeler and an arbitrageur trying to price and misprice assets, respectively. In the empirical estimation, the number of instruments used is $D = 3$, which means that the total number $N^{(\text{test})}$ of instrumented stock returns used is $3 \cdot 810 = 2,430$.

Feedforward Network

The second model considered is the prediction feedforward network developed by Gu et al. (2020). The FFN solves a conditional return prediction problem:

$$R^e_{t+1,i} = \mathbb{E}\left[R^e_{t+1,i}|\mathcal{I}_t\right] + \epsilon_{t+1,i} \tag{9}$$

where $\mu_{t,i} = \mathbb{E}_t\left[R^e_{t+1,i}\right]$ is the $t+1$ expected return of stock i conditional on information available at time t. The term $\epsilon_{t+1,i}$ is the forecast error. Following Gu et al. (2020), the information set $\mathcal{I}_t$ does not include any predictor lags or macroeconomic variables.[18] Equation (9) is enough to treat the problem as a machine learning problem. Specifically, the FFN learns the functional mapping from the firm characteristics to the next period returns; $\mu : \mathbb{R}^{K^{\text{char}}} \to \mathbb{R}$. As in Gu et al. (2020), the weighted sum of squared prediction errors is the empirical loss function to estimate the parameters of the FFN:

$$\hat{\mu}_{\text{FFN}} \in \arg\min_{\mu} \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N_t}\sum_{i=1}^{N_t}\left(R^e_{t+1,i} - \mu(\mathcal{I}_t)\right)^2 \tag{10}$$

where N_t is the number of stocks available in week t. As implied by Equation (2), current prices reflect investors' expectations of future asset payoffs. Because the FFN predicts these future payoffs, it is therefore closely related to the underlying asset pricing theory (Kelly and Xiu, 2023). Most importantly, in this predictive setting, the set of test assets is the same as the basis assets; that is $R^{(\text{basis})}_{t+1} = R^{(\text{test})}_{t+1}$. This means that the closed-form solution of the pricing kernel portfolio in Equation (5) applies. Therefore, $\mu_{t,i}$ must be proportional to the optimal weights $\omega^{\text{FFN}}_{t,i}$. This motivates the approximation $\mu_{t,i} \approx \omega^{\text{FFN}}_{t,i}$, which is also used by Gu et al. (2020). This preserves the cross-sectional ranking of the return predictions when constructing the portfolio weights. The weights $\hat{\omega}^{\text{FFN}}_{t,i}$ are also l_1-normalized weekly.

Elastic Net

This study uses a version of the elastic net asset pricing model in Kozak et al. (2020) ("KNS") adapted to individual stock returns. For a detailed overview of the general elastic net model of Zou and Hastie (2005), see Appendix A4. The original KNS model adds the penalty terms of the ridge and lasso models to a linear regression model. The regression uses the principal components of individual stock returns as test assets. For these PCA factors, they estimate the unconditional first and second sample moments; $\hat{\mu}$ and $\hat{\Sigma}$ and minimize a penalized version of the model specification error of Hansen and Jagannathan (1991):

[18] It is impossible to include lagged predictors here without significantly increasing network capacity

$$\hat{\omega}_{\text{KNS}} \in \arg\min_{\omega} \left[\left(\hat{\mu} - \hat{\Sigma}\omega \right)' \hat{\Sigma}^{-1} \left(\hat{\mu} - \hat{\Sigma}\omega \right) + 2\nu_1 \sum_{p=1}^{N^{(\text{PCA})}} |\omega_p| + \nu_2 \omega'\omega \right] \tag{11}$$

where v_1 and v_2 are the lasso and ridge penalty parameters, respectively.[19] As the KNS model uses PCA factors, but the GAN and FFN use individual stock returns, I adopt an alternative version of the KNS elastic net that is also considered in Chen et al. (2023a):

$$\hat{\theta}_{\text{EN}} \in \arg\min_{\theta} \left[\left(\frac{1}{T} \sum_{t=1}^{T} \tilde{F}_{t+1} - \frac{1}{T} \sum_{t=1}^{T} \tilde{F}_{t+1} \tilde{F}'_{t+1} \theta \right)^2 + 2\nu_1 \sum_{j=1}^{K^{\text{char}}} |\theta_j| + \nu_2 \theta'\theta \right] \tag{12}$$

Here, $\tilde{F}_{t+1} = \frac{1}{N} \sum_{i=1}^{N} I_{t,i} R^e_{t+1,i}$ is the K^{char}-dimensional vector of returns on characteristically managed portfolios. This estimation necessarily excludes macroeconomic information I_t. The portfolio weights follow as $\hat{\omega}_t^{\text{EN}} = \hat{\theta}'_{\text{EN}} I_{t,i}$ and are subsequently l_1-normalized. Note that different pairs of v_1 and v_2 will result in different sparse representations of the empirical pricing kernel M_{t+1}. Although I let the data choose the optimal pair, I do this considering an equivalent representation of the hyperparameters. That is, I estimate the total penalty $\alpha = \nu_1 + \nu_2$ and the l_1-ratio $\lambda = \nu_1/(\nu_1 + \nu_2)$. The estimation procedure uses the LARS-EN algorithm of Zou and Hastie (2005) in two sequential steps.[20] In the first step, a cyclical selection of feature updates is used to validate the hyperparameters α and λ on the validation set. Using the validated hyperparameters, a second estimation is performed using stochastic selection of features. This is required to produce five different estimates of θ_{EN}. Ultimately, these five estimates are treated as components of a final ensemble estimate.

Permanent-Transitory Decomposition and GAN

Alvarez and Jermann (2005) were the first authors to recognize that the pricing kernel M_{t+1} defined by Equation (4) could be decomposed into a permanent component $M^{(\text{P})}_{t+1}$ and a transitory component $M^{(\text{T})}_{t+1}$. This is called the permanent-transitory decomposition:

$$M_{t+1} = M^{(\text{P})}_{t+1} \cdot M^{(\text{T})}_{t+1} \tag{13}$$

Sufficient conditions for the existence of this decomposition are that the pricing kernel M_{t+1} and the test assets $R^{(\text{test})}_{t+1}$ are jointly stationary and ergodic (Alvarez and Jermann,

[19] In PCA space, the covariance matrix Σ is diagonal and the optimal weights become $\hat{\omega}_{\text{KNS},i} = \frac{\hat{\mu}_i - \nu_1}{\hat{\sigma}_i^2 + \nu_2}$ if $\hat{\mu}_i \geq \nu_1$ and 0 elsewhere (Lettau and Pelger, 2020)

[20] The Scikit-learn Python package is used to implement the LARS-EN

2005). In particular, the permanent component $M_{t+1}^{(\mathrm{P})}$ is a martingale; that is, $\mathbb{E}[M_{t+1}^{(\mathrm{P})}|\mathcal{I}_t] = 1$ almost surely.[21] The economic interpretation of the permanent pricing kernel is that it allows for long-term pricing (Hansen and Scheinkman, 2009). The transitory component $M_{t+1}^{(\mathrm{T})}$ corresponds to the reciprocal of the gross return on a portfolio of zero-coupon bonds (ZCB) with asymptotically long maturity. I contribute to the literature by extending the GAN model of Chen et al. (2023a) to include the permanent-transitory decomposition as an economic constraint. In particular, the PK network is decomposed into two sub-networks that construct the two components separately. The combined PK network links these networks by combining their output using the multiplicative permanent-transitory decomposition. In specifying each component network, two economically motivated priors from the theory are considered separately: (1) the permanent component is a martingale and (2) the transitory component is a portfolio of ZCBs. To this end, I introduce 3 new GAN model architectures that all share the same multiplicative decomposition, but varies in their implementation of these research priors. Specifically; (i) the Alvarez-Jermann GAN (AJ-GAN) ignores both priors, (ii) the AJ-Φ_ξ model imposes the martingale constraint on the permanent component, and (iii) the AJ$^{(\text{bond})}$ model builds the transitory component as a bond portfolio. All extended GAN models share the following unpenalized loss function:

$$\mathcal{L}^{(\text{unpen})}\left(\omega^{(T)}, \omega^{(P)}, Z \mid \mathcal{I}_t\right) = \frac{1}{N}\sum_{i=1}^{N} \frac{T_i}{T} \left\| \frac{1}{T_i}\sum_{t=1}^{T_i} M_{t+1}^{(\mathrm{P})} \cdot M_{t+1}^{(\mathrm{T})} R_{t+1,i}^{e} Z_t \right\|^2 \tag{14}$$

which is a combination of the permanent-transitory decomposition in Equation (13) and the baseline GAN loss function in Equation (7). The components are specified as follows:

$$M_{t+1}^{(\mathrm{P})} = 1 - \omega_t^{(\mathrm{P})\prime} R_{t+1}^{e} \quad \text{and} \quad M_{t+1}^{(\mathrm{T})} = 1 - \omega_t^{(\mathrm{T})\prime} R_{t+1}^{e} \tag{15}$$

The AJ-GAN model is the simplest extension of the baseline GAN model. The model naively applies the multiplicative decomposition of Equation (15) without considering the permanent component as a martingale or the transitory component as a portfolio of bonds. This model is included for two reasons. First, the idea with the AJ-GAN is to isolate the incremental effect of the decomposition on the ability of the GAN to perform in asset pricing tasks. Second, the AJ-GAN is the simplest possible generalization of the GAN model that also includes the permanent-transitory decomposition; see Figure II for the architecture of the AJ-GAN model. If this economic constraint is not helpful, the model has the flexibility to voluntarily reduce itself to the baseline GAN model. This would show up empirically in two

[21] Qin and Linetsky (2017) showed that in a no-arbitrage setting this decomposition exists under less restrictive assumptions if where the overall pricing kernel is a semimartingale.

ways; (1) the correlation of $M_{t+1}^{(\mathrm{T})}$ and $M_{t+1}^{(\mathrm{P})}$ would be perfect, meaning that the model wants to statistically reduce itself to a single pricing kernel component, or (2) the model could fix all predictions of one component to 1, letting the other proxy for the single pricing kernel.[22] The crucial drawback of the naive AJ-GAN model is that the permanent and transitory networks lose their economic interpretation. According to the theory, it is the martingale constraint on the permanent kernel or the bond-like properties of the transitory component that allow an econometrician to identify each component as either "transitory" or "permanent".

The second extension, the AJ-Φ_ξ model, uses the martingale constraint $\mathbb{E}_t[M_{t+1}^{(\mathrm{P})}] = 1$ to identify the permanent pricing kernel. Any further identification of the transitory component is ignored. In detail, a penalty term $\Phi_\xi(M_{t+1}^{(\mathrm{P})})$ is constructed as a regularization tool:

$$\Phi_\xi\left(M_{t+1}^{(\mathrm{P})}\right) = \xi \cdot \left(\frac{1}{T}\sum_{t=1}^{T} M_{t+1}^{(\mathrm{P})} - 1\right)^2 \tag{16}$$

where ξ is a scaling hyperparameter that determines the magnitude of the penalty to be imposed. The regularization term Φ_ξ is a scaled deviation from the conditional moment equation $\mathbb{E}[M_{t+1}^{(\mathrm{P})} - 1|\mathcal{I}_t] = 0$. The AJ-$\Phi_\xi$ model uses a penalized objective function $\mathcal{L}^{(\mathrm{pen})}$ which is the sum of the unpenalized loss function $\mathcal{L}^{(\mathrm{unpen})}$ from Equation (14) and the martingale regularization term Φ_ξ:

$$\mathcal{L}^{(\mathrm{pen})} = \mathcal{L}^{(\mathrm{unpen})} + \Phi_\xi\left(M_{t+1}^{(\mathrm{P})}\right) \tag{17}$$

The martingale constraint is not strictly binding in the sense that the permanent pricing kernel *must* obey this condition. Instead, adding it as a regularization term is a way for the researcher to express a research prior.[23] Although I ultimately let the data choose the optimal value of ξ, researchers could also use economically informed values of ξ to express varying degrees of conviction in the martingale constraint. Crucially, the AJ-GAN arises as a special case of the AJ-Φ_ξ model when $\xi = 0$.[24] Because of the multi-objective nature of the estimation, the AJ-Φ_ξ model can improve either in terms of pricing errors $\mathcal{L}^{(\mathrm{unpen})}$ or in terms of the martingale constraint $\Phi_\xi(M_{t+1}^{(\mathrm{P})})$. In particular, if the model finds a way to fix $\Phi_\xi(M_{t+1}^{(\mathrm{P})}) = 0$, the AJ-$\Phi_\xi$ model can update itself to find the optimal pricing kernel that perfectly satisfies the martingale constraint. The regularization term Φ_ξ also protects against local optima in $\mathcal{L}^{(\mathrm{unpen})}$. That is, the model can move out of the local optima by making

[22] For example, the model could set $M_{t+1}^{(\mathrm{P})} = 1$ for all t which makes $M_{t+1} = 1 \cdot M_{t+1}^{(\mathrm{T})}$ into the single tradable pricing kernel from the baseline GAN.

[23] Knowledge-based loss equations is a common way to integrate prior knowledge in the informed machine learning literature (Von Rueden et al., 2023)

[24] I specify Φ_ξ as a l_2-penalty to avoid the saturating gradient problem inherent in l_1-penalization.

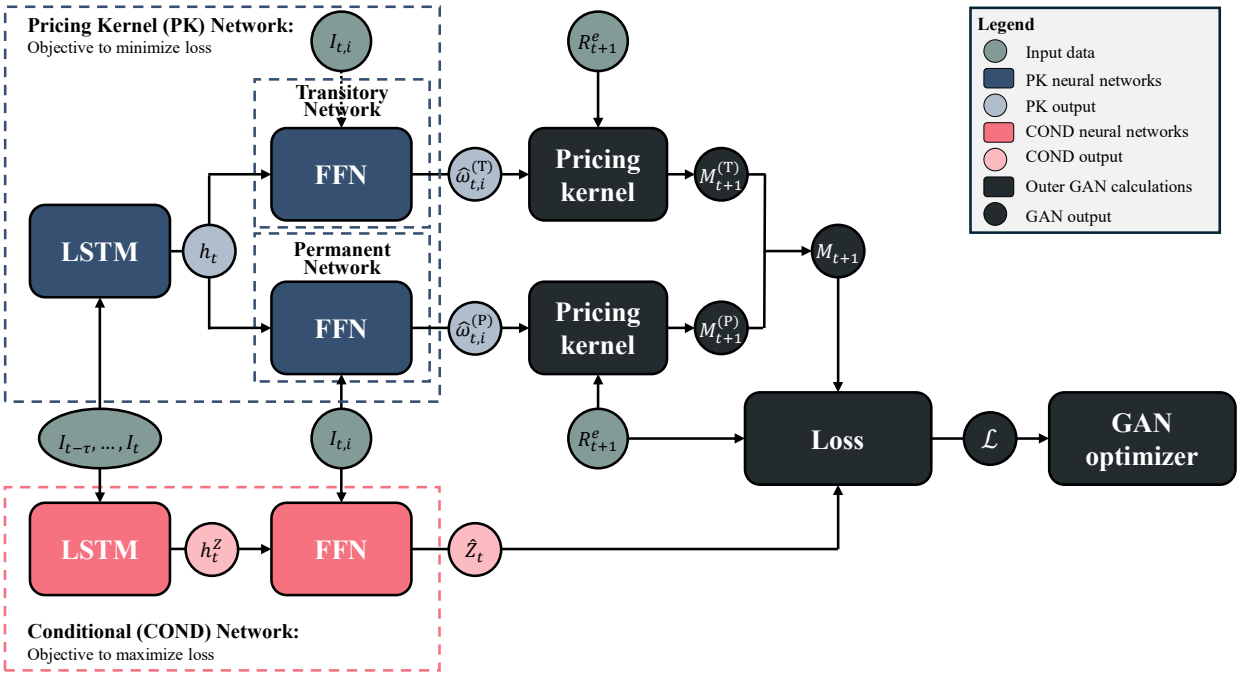


Figure II

Alvarez-Jermann GAN Model Architecture (AJ-GAN and AJ-Φ_ξ version)

Notes. This figure illustrates the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN). The model architecture updates the GAN of Chen et al. (2023a) by imposing the permanent-transitory decomposition as an economic constraint in the pricing kernel (PK) network. The PK network produces two portfolio weights $\omega_{t,i}^{(\mathrm{T})}$ and $\omega_{t,i}^{(\mathrm{P})}$ which are transformed by Equation (3) into permanent and transitory pricing kernels, respectively. That is, the transitory and permanent networks are produced by separate networks of machine learning models but are trained simultaneously with a joint objective function. The conditioning (COND) network produces moment conditions Z_{t+1} that translate into test assets. The COND network is specified exactly the same as in the baseline GAN model. Both the PK and COND networks use the macroeconomic information set I_t to produce separate hidden state estimates h_t and h_t^Z using long short-term memory (LSTM) cells (with lookback τ). These hidden states and firm characteristics $I_{t,i}$ are transformed by a feedforward neural network (FFN) into the final output of the networks. The outer GAN iterator combines the predictions with Nordic stock return data R_{t+1}^e to produce the objective function $\mathcal{L}$ of Equation (14). This loss measure is optimized by each network according to Algorithm (2).

improvements in Φ_ξ. The AJ-Φ_ξ model shares the same model architecture as the AJ-GAN.

Instead of identifying the permanent pricing kernel through the martingale constraint, the AJ$^{(\mathrm{bond})}$ model identifies on the transitory pricing kernel. Specifically, the transitory kernel is built as a bond portfolio and uses only bond characteristics $I_{t,b}^{(b)}$ and bond macroeconomic data $I_t^{(b)}$ as predictors. In detail, the transitory kernel is constructed as $M_{t+1}^{(\mathrm{T})} = 1 - \omega_t^{(\mathrm{T})} R_{t+1}^{e,(b)}$, where $R_{t+1}^{e,(b)}$ is the return vector for the Nordic bond indices. The idea is to let the transitory network build a dynamic bond portfolio that can proxy for the infinite-duration bond factor. Returns of shorter-duration bond indices are included among the basis assets to give the

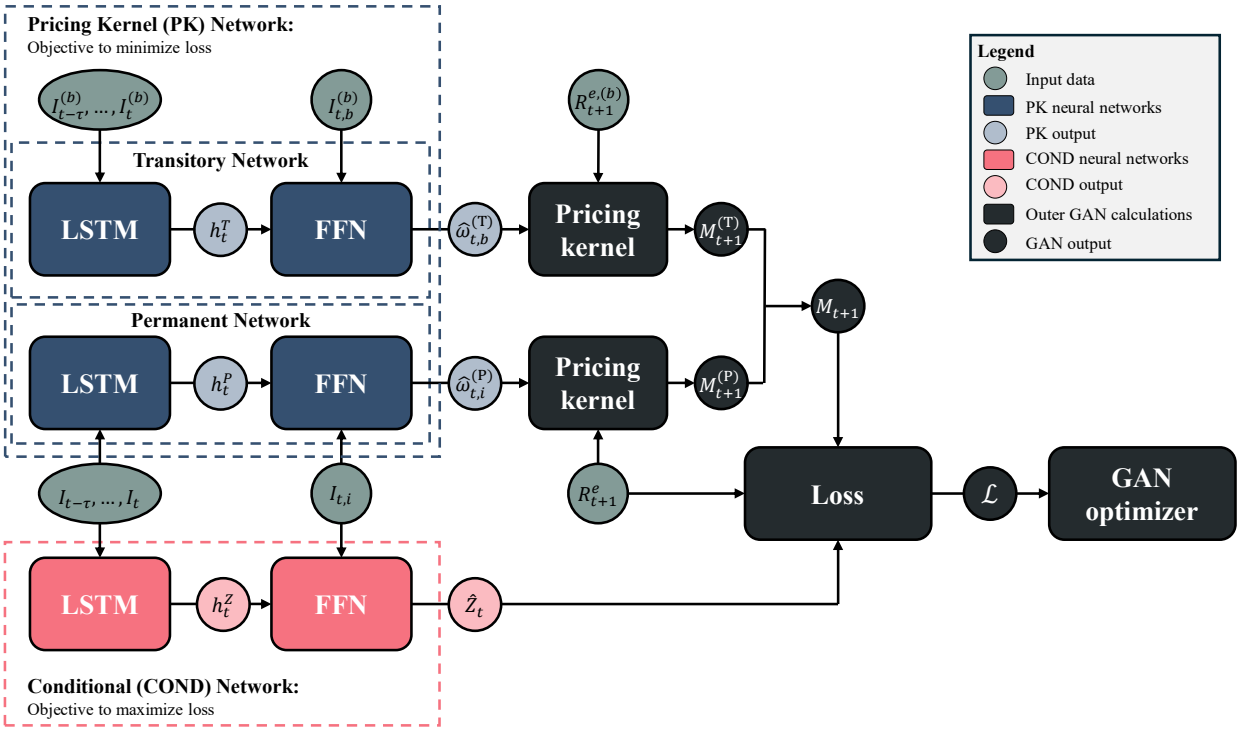


Figure III

Alvarez-Jermann GAN Model Architecture (AJ$^{(\text{bond})}$ version)

Notes. This figure illustrates the Alvarez and Jermann (2005) version of the generative adversarial network in which the transitory component is built using only bond data (AJ$^{(\text{bond})}$). The architecture updates the GAN of Chen et al. (2023a) by including the permanent-transitory decomposition as an economic constraint on the pricing kernel (PK) network. The PK network produces two portfolio weights $\omega_{t,b}^{(\text{T})}$ and $\omega_{t,i}^{(\text{P})}$ which are transformed by Equation (3) into permanent and transitory pricing kernels, respectively. That is, the transitory and permanent networks are produced by separate networks of machine learning models trained simultaneously with a joint objective function. The conditioning (COND) network produces instruments Z_{t+1} that are translated into test assets. The COND network is specified the same as in the baseline GAN model. Both the permanent and COND networks use the firm macroeconomic information set I_t to produce hidden state estimates h_t^P and h_t^Z using long short-term memory (LSTM) cells (with lookback τ). The transitory network uses its LSTM cell to transform the bond macroeconomic information set $I_t^{(b)}$ to its own hidden state h_t^T. The hidden states h_t^P and firm characteristics $I_{t,i}$ are transformed by the permanent network using a feedforward neural network (FFN) into weights (for stock i) in the permanent pricing kernel $\hat{\omega}_{t,i}^{(\text{P})}$. The transitory network use a FFN to build a bond portfolio $\hat{\omega}_{t,b}^{(\text{T})}$ (for bond index b) using its hidden state h_t^T and bond characteristics $I_{t,b}^{(b)}$. The outer GAN iterator combines the predictions with the returns on Nordic stocks R_{t+1}^e and Nordic bond indices $R_{t+1}^{e,(b)}$ to produce the objective function $\mathcal{L}$ of Equation (14). This objective is optimized by each network according to Algorithm (2).

pricing kernel additional flexibility. It might be necessary for the transitory network to build long-short strategies based on bond maturities to synthesize the return on the infinite-duration bond portfolio.

The AJ$^{(\text{bond})}$ model requires slight modifications to the AJ-GAN model architecture; see

Figure III. The new architecture completely separates the inputs used by the permanent and transitory networks. The permanent network uses its LSTM to transform the aggregate stock variables I_t into a permanent hidden state h_t^P. Likewise, the transitory network uses its LSTM to transform aggregate bond variables $I_t^{(b)}$ into the transitory hidden state h_t^T. The permanent network uses an FFN to transform the permanent hidden state and the firm characteristics $I_{t,i}$ into a weight in the permanent pricing kernel $\hat{\omega}_{t,i}^{(\mathrm{P})}$. Similarly, the transitory network uses an FFN to transform the transitory hidden state and bond characteristics $I_{t,i}^{(b)}$ into a weight within the transitory pricing kernel $\hat{\omega}_{t,b}^{(\mathrm{T})}$. Finally, the permanent component $M_{t+1}^{(\mathrm{P})}$ is a tradable factor built from stock returns R_{t+1}^e and the transitory component $M_{t+1}^{(\mathrm{T})}$ is a tradable factor built only from bond index returns $R_{t+1}^{e,(b)}$. Although the AJ$^{(\text{bond})}$ model uses bond returns $R_{t+1}^{e,(b)}$ as part of its basis assets, all test assets are still built using only stock returns. This means that the COND network is the same for the baseline GAN and all extended models. Therefore, potential improvements in the overall pricing kernel would be attributed to the new economic constraint and not to a higher capacity adversary. This allows for a fair side-by-side comparison between GAN models.

The empirical estimation procedure for the extended GAN models is detailed in Algorithm (2). This algorithm follows the same structure as the baseline Algorithm (1), but is adjusted for the new model architectures and loss functions. The extended GAN models share the same network structure; (i) one permanent pricing kernel (PK) network, (ii) one transitory PK network, and (iii) one conditional (COND) network. Because the transitory and permanent networks are part of the same PK network, this means that their parameters θ_T and θ_P are estimated simultaneously. The COND network is the same as in the baseline GAN model and has the same set of parameters φ. Although the permanent and transitory networks are separated, they are evaluated on the same objective function $\mathcal{L}$. This means that they can cooperate to beat the adversary in the zero-sum game. This "2 versus 1" architecture is unfair from the point of view of the adversary, but it is important to note that the regularization term Φ_ξ only handicaps the PK network.[25] For all extended GAN models, a synthetic non-tradable mean-variance optimal factor is extracted as $F_{t+1} = 1 - M_{t+1}$. In this setting, the decomposed factors $F_{t+1}^{(\mathrm{P})} = \omega_t^{(\mathrm{P})\prime} R_{t+1}^e$ and $F_{t+1}^{(\mathrm{T})} = \omega_t^{(\mathrm{T})\prime} R_{t+1}^e$ are tradable, but not necessarily mean-variance efficient on a standalone basis. However, the combined pricing kernel M_{t+1} should still be located on the Hansen and Jagannathan (1991) variance bound. For risk pricing purposes, a non-tradable factor is not a problem. The econometrician can still estimate time-varying risk loadings β_t with respect to the empirical non-tradable factor F_{t+1}.

[25] This is because updates in φ does not affect the martingale regularization term Φ_ξ

Risk Loadings

Given a function ω to estimate the PK weights, all that is required to perform risk pricing is a function $\beta_{t,i} = \beta(\mathcal{I}_t)$ to generate time-varying risk loadings. Following Chen et al. (2023a), the loadings for the GAN model are estimated using a simple FFN. This "GAN-Beta network" maps from the extracted hidden states $\hat{h}_t$ and firm characteristics $I_{t,i}$ onto the future product of the optimal return factor and the return of the stock; $F_{t+1}R^e_{t+1,i}$:[26]

$$\beta^{\text{GAN}} \in \arg\min_{\beta} \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N_t}\sum_{i=1}^{N_t}\left(F_{t+1}R^e_{t+1,i} - \beta(\mathcal{I}_t)\right)^2 \tag{18}$$

Estimating $\mathbb{E}_t[F_{t+1}R^e_{t+1}]$ in this way is akin to estimating the risk loadings up to a time-varying constant.[27] The ultimate goal is risk pricing in the cross-section. Therefore, this estimation procedure poses no problem since the cross-sectional ranking of the loadings at any point in time t is invariant to common time-varying constants. To keep the comparison as close as possible to the baseline GAN model, the corresponding "AJ-Beta", "AJ-Φ_ξ-Beta", and "AJ$^{(\text{bond})}$-Beta" networks all share the same model architecture, estimation procedure, and objective function as the validated GAN-Beta network.

The FFN loadings are based on the conditional beta representation from Equation (2):

$$\mathbb{E}_t\left[R^e_{t+1,i}\right] = \left(-\frac{\mathbb{C}\text{ov}_t\left(R^e_{t+1,i}, M_{t+1}\right)}{\mathbb{V}\text{ar}_t(M_{t+1})}\right)\cdot\frac{\mathbb{V}\text{ar}_t(M_{t+1})}{\mathbb{E}_t[M_{t+1}]} = \beta_{t,i}\mathbb{E}_t\left[F_{t+1}\right] \tag{19}$$

This relationship is also called the security market line and suggests that the expected excess return of an asset is perfectly linear to the expected risk premium of the mean-variance optimal factor F_{t+1}. The theory suggests that the relationship should hold with a zero intercept. Therefore, the predictions of the FFN model $\mu_{t,i} = \mathbb{E}_t[R^e_{t+1}]$ are linearly related to the loadings $\beta_{t,i}$. Crucially, the cross-sectional ranking of μ and β will be the same within each week. This motivates the approximation $\mu_{t,i} \approx \beta_{t,i}$, which is also used by Gu et al. (2020). The loadings β_{EN} of the elastic net are estimated by a linear projection of $R^e_{t+1}F_{t+1}$ onto $I_{t,i}$.[28] In a post-processing step, the loadings of all models are scaled so that the beta of the empirical pricing kernel is always 1; $\forall t : \beta_t'\omega_t = 1$. Because the extended GAN models produce non-tradable factors F_{t+1}, no scaling of the AJ-GAN, AJ-Φ_ξ, and AJ$^{(\text{bond})}$ loadings is performed. This means that the economic magnitude of the loadings loses their interpretation. However, cross-sectional sorting of the loadings can still be performed for risk

[26] This objective avoids the measurement problem in estimating $\mathbb{E}_t[R^e_{t+1,i}]$ (Chen et al., 2023a)

[27] This is evident from the covariance formula, i.e. $\mathbb{C}ov(F_{t+1}, R^e_{t+1,i}) = \mathbb{E}_t[F_{t+1}R^e_{t+1}] - \mathbb{E}_t[F_{t+1}]\mathbb{E}_t[R^e_{t+1}]$.

[28] This is the linear analogue to the nonlinear problem in Equation (18) solved by the GAN-Beta network.

pricing purposes. Finally, the scaled loadings are winsorized at ±4.

Evaluation Metrics

The asset pricing performance metrics considered in this study are defined in the same way as in Chen et al. (2023a). In the no-arbitrage setting, a good empirical pricing kernel should be on the volatility bound of Hansen and Jagannathan (1991). Crucially, the return factor F_{t+1} should achieve the maximum attainable Sharpe ratio (SR):[29]

$$\text{SR} = \frac{\mathbb{E}\left[R^e_{t+1}\right]}{\sigma\left(R^e_{t+1}\right)} \tag{20}$$

Comparing the out-of-sample Sharpe ratio of all models and the market-weighted portfolio of stocks allows us to answer RQ 1 and RQ 2. Additionally, the Sharpe ratio is used to answer part (i) of RQ 8 by comparing this metric for the extended GAN models with the baseline GAN model. That is, does the permanent-transitory constraint lead to improvements in the signal-to-noise ratio? The predictive relationship in Equation (19) can be used to empirically test the predictive ability of an asset pricing model and to answer part (i) of RQ 3. The predictive power of the model is evaluated using the R^2 of the linear relationship between the expected return and the loadings, and on whether the empirical intercept is close to zero. The time-varying nature of the loadings $\beta_{t,i}$ means that individual stocks need to be aggregated into portfolios. By decile-sorting stocks based on their loadings, the aggregated portfolios will have more stable loadings over time. The returns of the portfolios are projected onto the return of the market-weighted portfolio. The intercepts of this CAPM regression correspond to pricing errors and are used to perform the GRS asset pricing test of Gibbons et al. (1989). In this context, the GRS test has the null hypothesis that the CAPM correctly prices all generated decile-sorted portfolios.[30] This test directly answers whether the CAPM model already capture the signals generated by of the models (RQ 4). In the no-arbitrage framework, excess returns can be projected onto the loadings β_t and subsequently residualized:

$$\hat{\epsilon}_{t+1} = \left(I_N - \hat{\beta}_t(\hat{\beta}_t'\hat{\beta}_t)^{-1}\hat{\beta}_t\right) R^e_{t+1} \tag{21}$$

Where I_N is the N-dimensional identity matrix. This residualization procedure decomposes excess returns into a systematic signal and an idiosyncratic noise: $R^e_{t+1,i} = \beta_{t,i}F_{t+1} +$

[29] In asset pricing, the Sharpe ratio is synonymous with "signal-to-noise ratio" (Nagel, 2021)

[30] Specifically, I use the empirical Sharpe ratio $\text{SR}^2_{\text{CAPM}}$ of the market, the estimated vector of pricing errors $\boldsymbol{\alpha}$, and covariance matrix of residuals $\boldsymbol{\Omega}$ to perform the χ^2 test: $T(1+\text{SR}^2_{\text{CAPM}})^{-1}\boldsymbol{\alpha}'\boldsymbol{\Omega}^{-1}\boldsymbol{\alpha} \sim \chi^2_{10}$

$\epsilon_{t+1,i}$ (Chen et al., 2023a). Using the estimates of the residuals $\hat{\epsilon}_{t+1}$, I create two metrics of explained variation (EV) to assess part (ii) of RQ 3. The first is a time series R^2 metric:

$$\text{TS-}R^2 = 1 - \frac{\frac{1}{T}\sum_{t=1}^{T}\frac{1}{N_t}\sum_{i=1}^{N_t}(\hat{\epsilon}_{t+1,i})^2}{\frac{1}{T}\sum_{t=1}^{T}\frac{1}{N_t}\sum_{i=1}^{N_t}(R^e_{t+1,i})^2} \tag{22}$$

This metric measures how much variation in individual stock returns is explained by the models. The second metric is a cross-sectional measure of explained variation:

$$\text{CS-}R^2 = 1 - \sum_{i=1}^{N}\hat{\alpha}_i^2 = 1 - \frac{\frac{1}{N}\sum_{i=1}^{N}\frac{T_i}{T}\left(\frac{1}{T_i}\sum_{t=1}^{T_i}\hat{\epsilon}_{t+1,i}\right)^2}{\frac{1}{N}\sum_{i=1}^{N}\frac{T_i}{T}\left(\frac{1}{T_i}\sum_{t=1}^{T_i}R^e_{t+1,i}\right)^2} \tag{23}$$

Which is related to the empirical pricing errors $\hat{\alpha}$ produced by the model.[31] This metric measures how much variation in the cross section of average returns is explained by a model. Because asset pricing makes predictions about expected returns and not individual returns, an informed prior is that the TS-R^2 metric of a good asset pricing model could be close to zero, but CS-R^2 should be significantly positive. The TS-R^2 and CS-R^2 are also used to answer part (ii) of RQ 8. That is, does the permanent transitory constraint improve the ability of the GAN model to explain individual and expected returns? As a robustness check, these explained variation metrics are also evaluated on 460 decile-sorted and value-weighted portfolios for each firm characteristic. The portfolios will average out some of the idiosyncratic noise in individual returns, which allows the models to improve their explanatory power. In addition, portfolios aggregate information, which will strengthen characteristic-related signals. For example, if a higher or lower book-to-market ratio is a meaningful predictor of future returns, then aggregating stocks into portfolios by deciles of book-to-market should lead to noticeable differences in return dynamics. If the result for RQ 3 is robust, the GAN, FFN, and EN should be ranked the same in terms of TS-R^2, CS-R^2, and pricing errors α on these portfolios. The same goes for the extended GAN models and part (ii) of RQ 8.

A saliency analysis is performed to rank the characteristics according to their importance. This analysis will answer whether market frictions drives the performance of the models (RQ 5 and part (i) of RQ 9) and which characteristics are the most important for the pricing of Nordic stocks (RQ 6). This is done by measuring the sensitivity of the predictions to perturbations in the input predictors. Specifically, the variable importance (VI) for each characteristic $C_{t,i} \in I_{t,i}$ is calculated as the normalized average absolute gradient:

[31] This specification scales the pricing errors by the root-mean-square of all average returns

$$\text{VI}(C_{t,i}) = \frac{1}{C} \sum_{i=1}^{N} \sum_{t=1}^{T} \left| \frac{\partial \hat{\omega}(\mathcal{I}_t)}{\partial I_{t,i}} \right| \tag{24}$$

where C is a normalization constant to make the sum of variable importance across all characteristics equal to one. This is a generalized version of a linear regression coefficient (Chen et al., 2023a). Therefore, the unscaled VI of the elastic net model is the same as θ_{EN}. The elastic net is particularly adept at selecting predictors, as it is explicitly specified to perform variable selection (the l_1-penalty) and to shrink the influence of the least informative characteristics (the l_2-penalty). Using the most important variables from this saliency analysis, pairwise interactions of predictors are visualized to investigate the prevalence of nonlinear patterns. This analysis will answer RQ 7 and part (ii) of RQ 9.

Model Architecture and Hyperparameter Selection

Because machine learning models are highly parameterized, they have to be regularized to reduce overfitting (Goodfellow et al., 2016). To this end, the elastic net penalties implement $l1$- and $l2$-shrinkage to reduce the impact of uninformative predictors. Instead, the FFN, GAN, and extended GAN models use a machine learning technique called dropout. The method applies a stochastic mask to drop a preset number of nodes during training. For a detailed overview of dropout in neural networks, see Appendix A4. The effect of dropout is to reduce the tendency of the model to memorize the training examples. This reduces the risk of overfitting and poor out-of-sample performance.

The architecture of a machine learning model depends on the hyperparameters of the model, for example, the number of layers or the learning rate of a neural network. There is no universal machine learning algorithm that performs the best for all tasks.[32] As a result, the hyperparameters have to be validated using a data-driven validation procedure before the final empirical estimation is performed. In this study a grid of possible hyperparameter configurations is considered, and the validated hyperparameters are those associated with the highest signal-to-noise ratio on the validation set.[33] For the GAN-Beta network, the validated model is the one with the highest TS-R^2. To be comparable to the baseline GAN model, extended GAN models use the same architecture as the validated baseline GAN model. In detail, the AJ-GAN and AJ-Φ_ξ models use the exact same LSTM configurations and number of hidden states h_t as the baseline model. Because the AJ$^{(\text{bond})}$ model has two LSTM cells, the number of hidden states h_t produced per LSTM cell is reduced by 50%.[34] The FFN cells

[32] This is the no free lunch (NFL) theorem in statistical learning by Wolpert and Macready (1997).

[33] While the grid is coarse its informed by grid points used in Chen et al. (2023a) and Gu et al. (2020)

[34] The validated number of states was 2 (Table A12), so the AJ$^{(\text{bond})}$ LSTM cells produce 1 state each

in the extended GAN models share the same configurations as the validated FFN cell in the baseline model. Similarly, the Beta network of the extended GAN models uses the same configurations as the validated GAN-Beta network. Details of the grid used, and the results of the validation procedure are presented in Appendix A6.

Once the models have been validated, the functions ω_t and β_t are estimated 5 times using different seeds and combined into a single ensemble model. Ensemble learning can improve the variance-bias trade-off by making the combined model less dependent on a single seed used for initialization. It also reduces the overall impact of getting stuck at a local minima within any one training process.[35] The five model estimates are treated as components of the final ensemble weights $\omega_{t,i}$ and the GAN loadings $\beta_{t,i}^{\text{GAN}}$, respectively. In detail, the final estimates are aggregated by the mean: $\hat{\omega} = \frac{1}{5}\sum_{m=1}^{5} \hat{\omega}_{(s_m)}$ and $\hat{\beta}^{\text{GAN}} = \frac{1}{5}\sum_{m=1}^{5} \hat{\beta}_{(s_m)}^{\text{GAN}}$. Here s_m denotes the initial seed used for model m.[36] Finally, the ensemble estimates are evaluated on the test set according to the performance metrics defined in the previous section.

IV. Empirical Results for Nordic Equities

IV.A Asset Pricing Performance

To summarily answer RQ 1 and RQ 2; the GAN is associated with an out-of-sample signal-to-noise ratio of 2.41 compared to 1.29 and 1.90 for the FFN and EN, respectively; see Table I. In particular, all models produce a less noisy signal than the benchmark CAPM model; see Figure IV for time series of all pricing kernels.[37] All models also produce higher signal-to-noise ratios than the Fama and French (1992) 3-factor model and the Fama and French (2015) 5-factor model; see Appendix A7. Although the GAN, FFN, and EN have similar Sharpe ratios in-sample, the out-of-sample Sharpe ratio clearly separates. The no-arbitrage constraint and robust control inherent in the adversarial estimation procedure of the GAN allow it to produce a more generalizable signal. The fact that the elastic net has a higher Sharpe ratio than the FFN reflects the high asset pricing performance attainable by sparse linear factor models. Furthermore, the Sharpe ratios of the ensemble GAN and FFN models are larger than the mean of their parts; see Figures A17 and A24. These economic gains underline the importance of ensemble learning in empirical asset pricing. In comparison, ensemble learning does not produce comparable gains for the EN; see Table A13. All models

[35] Ensemble learning is motivated by the wisdom of crowds phenomenon, i.e. aggregating many weak learners (see Schapire, 1989) can produce a model with strong predictive ability

[36] As in Chen et al. (2023a), instruments Z and hidden states h are not aggregated due to a lack of consistent interpretation across ensemble components.

[37] The value-weighted market factor had Sharpe Ratios of 0.05 (train), 1.32 (valid), and 0.72 (test)

Table I
Performance of Empirical Pricing Kernel Models

	Sharpe Ratio			Time-Series R^2			Cross-Sectional R^2		
Model	Train	Valid	Test	Train	Valid	Test	Train	Valid	Test
EN	3.23	3.31	1.90	0.01	0.00	0.00	0.72	0.57	0.42
FFN	3.32	2.93	1.29	0.03	0.02	0.03	0.69	0.63	0.47
GAN	3.82	4.73	2.41	0.08	0.06	0.10	0.77	0.73	0.63
AJ-GAN	3.77	4.69	2.13	0.12	0.08	0.14	0.77	0.74	0.66
AJ-Φ_ξ	3.82	4.66	2.55	0.11	0.07	0.13	0.76	0.74	0.65
AJ$^{(\text{bond})}$	5.05	5.15	2.85	0.14	0.08	0.15	0.77	0.74	0.66

Notes. The table reports the answers to research questions RQ 1, part (ii) of RQ 3, and RQ 8. For each sub-sample; the performance metrics of the feedforward neural network (FFN), elastic net (EN), and three extended versions of the generative adversarial network (GAN) are presented. The Alvarez and Jermann (2005) version of the GAN model (AJ-GAN) applies a naive decomposition of the pricing kernel; $M_{t+1} = M^{(\text{T})}_{t+1} \cdot M^{(\text{P})}_{t+1}$. The AJ-$\Phi_\xi$ model extends the AJ-GAN by adding the martingale penalty term Φ_ξ to the objective function. AJ$^{(\text{bond})}$ is an alternative to the AJ-GAN model in which the transitory pricing kernel $M^{(\text{T})}_{t+1}$ is built using only bond data and the permanent pricing kernel $M^{(\text{P})}_{t+1}$ is built using only stock data. Performance metrics include the annualized Sharpe ratio (SR), explained variation of individual returns (TS-R^2) and explained variation in average returns (CS-R^2).

see a decrease in the signal-to-noise ratio on the transition from in-sample to out-of-sample data. This could imply a degree of overfitting in the training process, but this is not apparent from Figures A17 and A24. Alternatively, underlying structural breaks in the Nordic financial market could be a reason for this result. The GAN might be better able at capturing regime changes because it directly estimates hidden states; see Figure A5 for estimates of h_t. This could explain why the GAN is particularly quick to recover from the 2020 Covid crash. To answer part (ii) of RQ 3; all models have some explanatory power in the average returns, but only the GAN has a meaningful explanatory power in individual returns. The GAN explains 10% of individual stock returns and 63% of average returns. A TS-R^2 of 10% is very high in an asset pricing setting, and the FFN and EN explain only 3% and 0% of individual returns and 47% and 42% of average returns, respectively. The high Sharpe ratio and CS-R^2 but low TS-R^2 indicates that the selection and shrinkage performed by the EN successfully capture systematic risks while ignoring most of the idiosyncratic effects.

The extended GAN models all produce signal-to-noise ratios higher than those of the FFN and EN models. However, the AJ-GAN model does not produce a higher Sharpe ratio compared to the baseline GAN model, but it does yield a significant improvement in explanatory power in both individual and average returns. The AJ-GAN explains about 14%

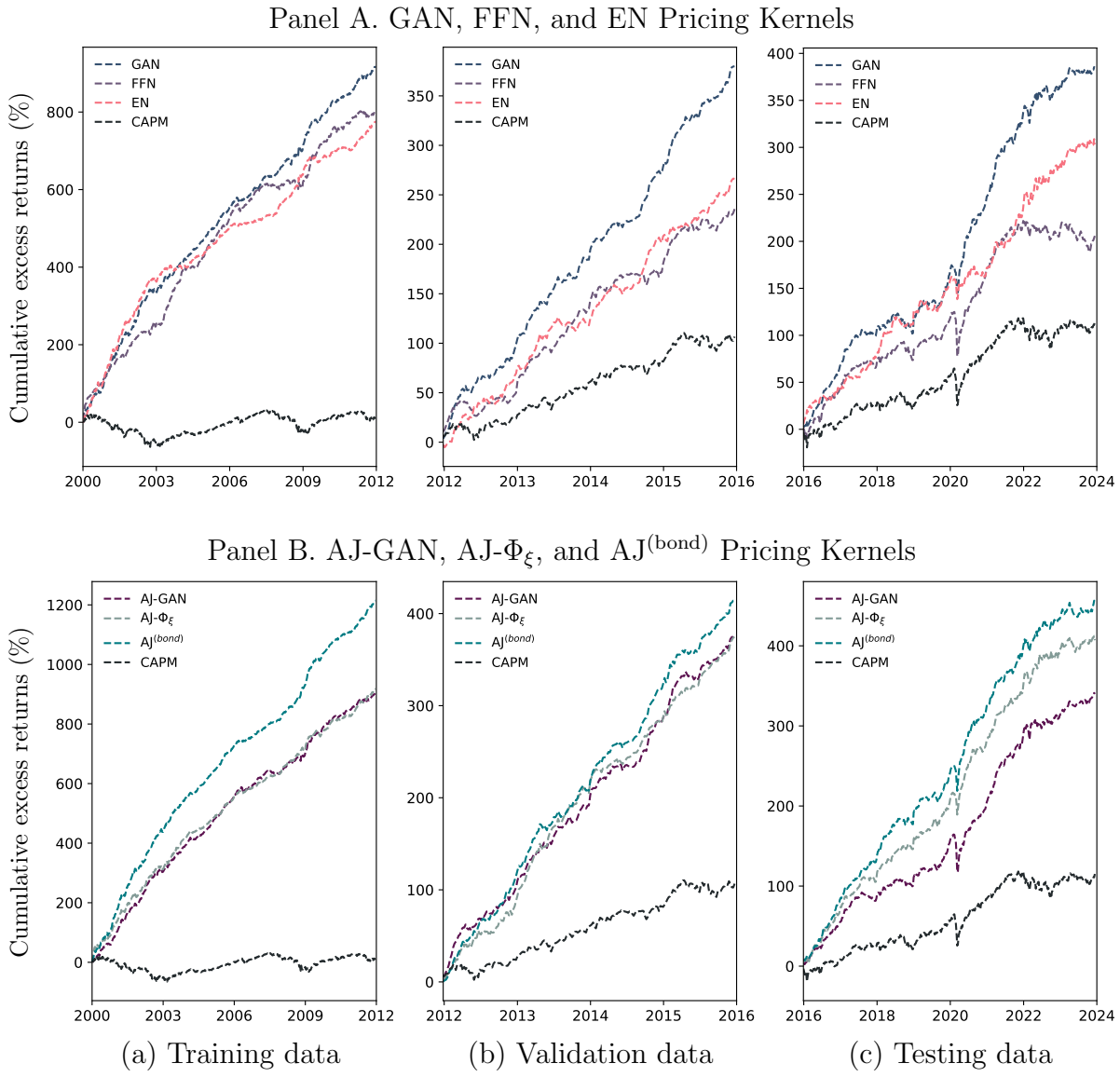


Figure IV
Time Series of Estimated Pricing Kernels

Notes. This figure answers RQ 2 by illustrating the empirical cumulative excess returns of each pricing kernel. Panel A illustrates these time series for the generative adversarial network (GAN), the feedforward network (FFN), and the elastic net (EN). Panel B illustrates the empirical pricing kernels of the Alvarez and Jermann (2005) version of the GAN model (AJ-GAN), the AJ-GAN including the martingale constraint (AJ-Φ_ξ), and the AJ-GAN with a temporary component built using only bond data (AJ$^{(\text{bond})}$). The CAPM illustrates the return on the value-weighted portfolio of all stocks in the sample at time t. This corresponds to the market factor of the opportunity set and is rebalanced prior to each week. The cumulative return is reset to zero on the transition of each sub-sample to the next, i.e. from training data to validation data and from validation data to testing data. All time series are scaled to have an annualized volatility of 20%.

of individual returns and 66% of average returns. This improvement in explanatory power is common in all extended GAN models. The AJ-Φ_ξ and AJ$^{(\text{bond})}$ models are associated with TS-R^2 of 13% and 15%, respectively. Similarly, the extended models all explain more of the

cross-section of average returns. The AJ-Φ_ξ and AJ$^{(\text{bond})}$ models explain 65% and 66% of expected returns, respectively, which is a 3 percentage point increase to the baseline GAN. Interestingly, the signal-to-noise ratio of the AJ-GAN model improves from 2.13 to 2.55 when the martingale constraint is imposed. The penalty term Φ_ξ allows the model to better identify the permanent component. At the beginning of the training process, the penalty term contributes about 30-40% to the objective of the AJ-Φ_ξ model; see Figure A19. At the end of the training period, the regularization term contributes almost nothing, suggesting that the martingale constraint was successfully imposed in-sample. In contrast, the naive decomposition of the AJ-GAN leaves it unable to separate the signal-to-noise ratios of its pricing kernel components; see Figure A23. At the same time, the AJ-Φ_ξ model clearly separates the transitory component from the permanent component, allowing them to converge to different Sharpe ratios out-of-sample. The martingale constraint Φ_ξ makes the empirical pricing kernels behave more in line with the theoretical implications of Alvarez and Jermann (2005) and Hansen and Scheinkman (2009). Although the martingale regularization for the permanent component is economically meaningful, the most meaningful economic constraint is to treat the transitory component as a bond portfolio. Expanding the information set and basis assets to include Nordic bond indices allows the AJ(bond) model to distinguish itself from the other models. It performs the best in all evaluation metrics on all subsamples. It achieves a Sharpe ratio of 5.05 in-sample, 5.15 on the validation set, and 2.85 out-of-sample, which all are economically meaningful improvements compared to both the baseline GAN model and the other extended GAN models. The AJ$^{\text{bond}}$ explains about 15% of individual returns and 66% of average returns. The fact that the improvement in explanatory power is similar for all extended GAN models suggests that imposing the permanent-transitory decomposition is associated with a level shift in explanatory power. This level shift is partially attributed to the imperfect correlation of the pricing kernel components. In line with the long-run risk model, but contrary to previous empirical results in Bansal and Yaron (2004), the correlation of the transitory and permanent components within the AJ-GAN and AJ-Φ_ξ models is consistently *negative* across model fits, see Figure A20 and Figure A21. This negative correlation means that each pricing kernel component explains very different return dynamics and reduce the variance of the combined pricing kernel when considered together. In contrast, the correlation of the components within the AJ$^{(\text{bond})}$ model is close to zero throughout the entire training process, see Figure A22. The persistence of these imperfect correlations suggest that neither extended GAN model wants to reduce itself to a single pricing kernel, as in the baseline case. This effect shows up empirically as very different time series of the kernels (see Figure A1) and different correlations with respect to the market factor (see Table A1). Finally, Table A20, Table A21, and Table A22 illustrate the variance ratio of the pricing kernel

components. Prior literature would lead us to believe that the permanent component should have significantly higher variance than the transitory component (Backus et al., 2014). The AJ(bond) model achieves this result easily as most of the variance comes from the permanent component. In contrast, the AJ-GAN and AJ-Φ_ξ models produce transitory and permanent components with about the same variance. The AJ(bond) model also produce very different hidden states compared to the other GAN models. The permanent hidden state h_t^P is similar to the other models but the transitory hidden state h_t^T produces very different dynamics, see Figure A8. This suggest that building one hidden state using only stock data and one hidden state using only bond data can allow the GAN to capture different dynamics of the macroeconomic environment.

The fact that all extended GAN models outperform the baseline GAN in terms of explained variation suggests that the answer to part (ii) of RQ 8 is positive. That is, adding the permanent-transitory constraint can provide significant economic gains in terms of explanatory power on its own. The answer to part (i) of RQ 8 is more nuanced. The fact that the AJ-Φ_ξ and AJ$^{(\text{bond})}$ models outperform the baseline in terms of the Sharpe ratio suggests that economic gains in the signal-to-noise ratio require careful use of additional economic constraints. Such additional economic constraints could be the martingale constraint or the construction of the transitory component as a bond portfolio. A naive decomposition will not improve the signal-to-noise ratio on its own, as evidenced by the signal-to-noise ratio of the AJ-GAN model. These findings corroborate the sentiment of Nagel (2021) and Avramov et al. (2023). That is, adding economically motivated constraints to machine learning models can improve the model's ability to perform in an asset pricing setting.

IV.B Predictive Performance

Figure V illustrates the results to part (i) of RQ 3. The GAN, FFN, and EN all produce an empirical security market line that is apparently linear. This predictive relationship achieves c.a. 95% R^2 in every case. Ultimately what differentiate the models are the intercepts. Only the GAN, with an intercept of 0.97%, is close to the zero intercept expected from the theory. The FFN and EN require intercepts of much larger magnitude just to achieve a R^2 on par with the GAN. This indicates that the no-arbitrage constraint inherent in the estimation of the GAN model brings the model closer in line with asset pricing theory. Although the R^2 of the SML starts lower in the training data for the GAN than that of the FFN and EN, it improves from 0.94 to 0.95 when going from in-sample to out-of-sample. In contrast, the R^2 of the FFN and EN deteriorates from 0.99 to 0.94 and 0.98 to 0.95, respectively. This implies that the predictive power of the FFN and EN models is less robust. This corroborates

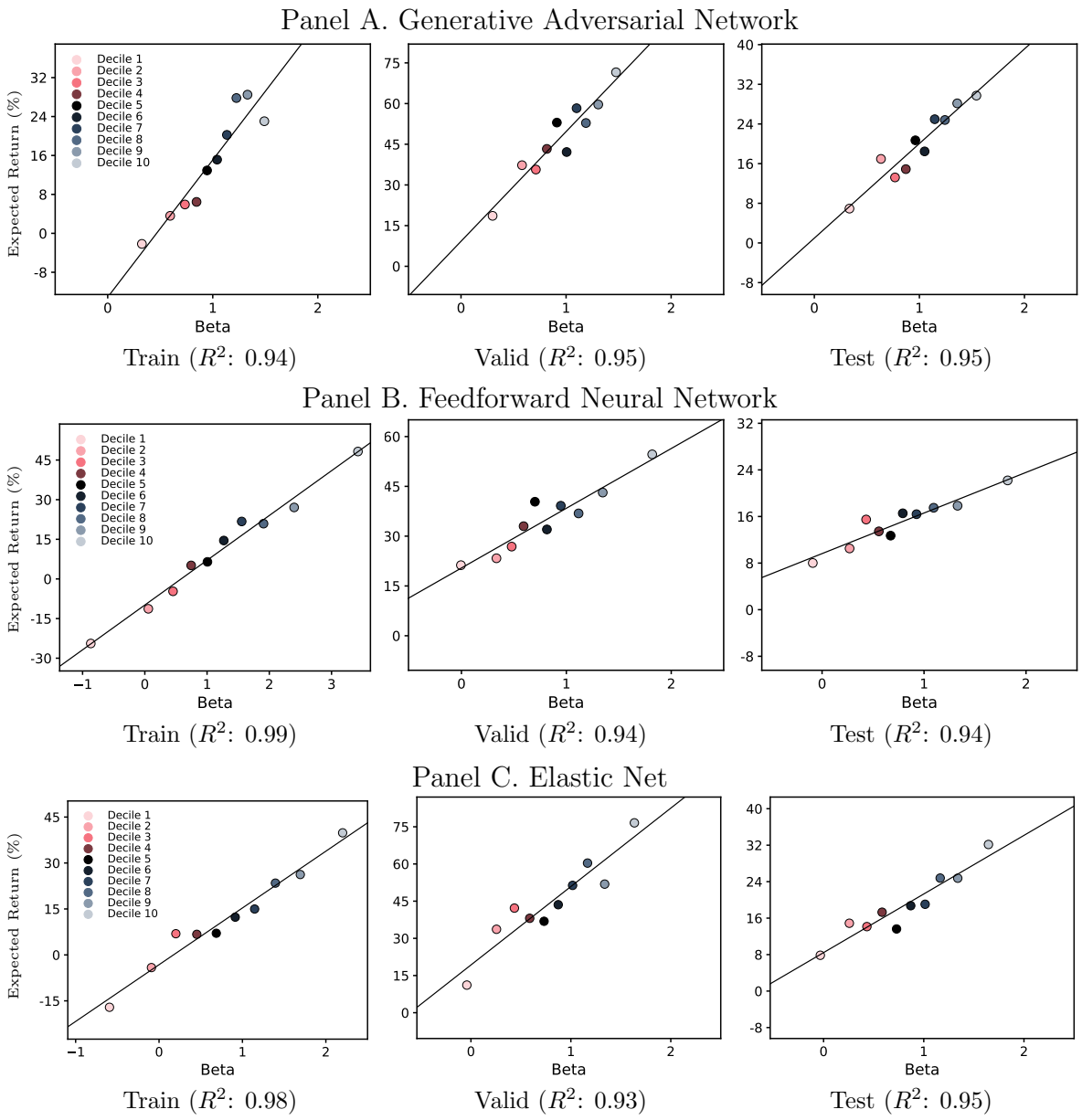


Figure V

Beta Representation of 10 β-sorted Portfolios

Notes. This figure answers part (i) of RQ 3 by illustrating the weekly beta representation $\mathbb{E}_t[R^e_{t+1,i}] = \beta_{t,i}\mathbb{E}_t[F_{t+1}]$ of all models. Panel A - Panel C present the results for the generative adversarial network (GAN), the feedforward network (FFN), and the elastic net (EN), respectively. Within the panels, the security market line (SML) is plotted for the training, validation, and testing datasets (from left to right). The unadjusted R^2 is presented for each model and subset. Portfolios are equally weighted based on deciles of the loadings $\beta_{t,i}$ generated by each model, and are rebalanced weekly. The unconditional average of the portfolio returns is annualized and expressed as a percentage. The linear regression includes an intercept.

the notion of the adversary as a tool for robust control. The GAN finds not only a more general portfolio representation $\omega_{t,i}$ for the pricing kernel, but also for the risk loadings $\beta_{t,i}$. Visually, Figure A2, Figure A3, and Figure A4 illustrate the model variance more clearly.

The performance of the β-sorted portfolios is clearly separable in-sample, but is slightly less distinguishable on out-of-sample data.

Table II
Pricing Performance on Decile Sorted β-Portfolios versus CAPM

	Test Data (2016 - 2023)											
	GAN				FFN				EN			
Decile	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p
1	6.9	-8.5	-1.84	0.07*	8.0	-4.9	-1.53	0.13	7.9	-7.9	-1.84	0.07*
2	17.0	0.6	0.17	0.87	10.5	-2.7	-0.92	0.36	14.9	-1.1	-0.27	0.79
3	13.2	-3.3	-0.91	0.36	15.5	2.3	0.80	0.43	14.1	-1.8	-0.44	0.66
4	14.9	-1.2	-0.28	0.78	13.4	0.3	0.09	0.93	17.3	1.2	0.30	0.77
5	20.7	4.5	1.16	0.25	12.7	-0.4	-0.15	0.88	13.6	-2.4	-0.58	0.56
6	18.5	2.4	0.60	0.55	16.5	3.5	1.15	0.25	18.7	2.9	0.68	0.50
7	25.0	9.4	2.10	0.04**	16.4	3.7	1.10	0.27	19.0	3.1	0.75	0.46
8	24.8	9.8	1.99	0.05**	17.5	5.2	1.40	0.16	24.8	9.0	2.09	0.04**
9	28.2	13.5	2.61	0.01***	17.8	6.2	1.48	0.14	24.8	9.1	2.06	0.04**
10	29.7	16.3	2.75	0.01***	22.2	11.4	2.42	0.02**	32.2	16.9	3.56	0.00***
10-1	22.8	24.8	3.65	0.00***	14.2	16.3	3.53	0.00***	24.3	24.8	5.05	0.00***
GRS AP Test			χ^2_{10}	p			χ^2_{10}	p			χ^2_{10}	p
			21.6	0.017**			18.0	0.056*			35.4	0.000***

Notes. This table reports the results to RQ 4. Each model generates risk loadings $\beta_{t,i}$ for all stocks. These loadings are sorted by deciles every week, and equally weighted portfolios are constructed within deciles. The sample estimate of the unconditional expected return for each portfolio is annualized and presented under the heading $\mathbb{E}[R^e_{t+1}]$. The annualized pricing error (α), its t-stat (t) and the two-tailed p-value (p) are calculated from a projection of portfolio returns onto the return of the market-weighted portfolio. Excess returns and pricing errors are expressed as percentage units. The GRS asset pricing (AP) test statistic is defined in Gibbons et al. (1989) and has the null hypothesis that the CAPM correctly prices all decile portfolios. Newey-West standard errors are used to correct for heteroskedasticity and autocorrelation (using $T^{1/4} \cdot 0.75 \approx 2$ lags). Significance levels are specified as follows: * $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$.

To answer RQ 4, Table II reports out-of-sample asset pricing tests for decile-sorted portfolios based on risk loadings. The benchmark presented here is the Capital Asset Pricing Model of Sharpe (1964), but the results are robust to the Fama and French (1992) 3-factor model and the Fama and French (2015) 5-factor model; see Appendix A7. For individual deciles of the GAN and EN, only the pricing errors α of the most extreme positive deciles are significant at the 1% level. In comparison, none of the deciles of the FFN model is significant at the 1% level. This is related to the apparent wedge in average returns on the highest deciles when comparing the FFN with the GAN and EN. The CAPM does a surprisingly good job of pricing the generated decile portfolios, even though its own signal-to-noise ratio is much lower. At the 10% level, mispricings for 5 of 10 GAN portfolios and 4 of 10 EN portfolios would be considered statistically significant, but only 1 portfolio for the FFN. Still, the lowest p-values are clearly biased towards the highest positive decile portfolios.

Table III
Pricing Performance on Decile Sorted AJ-β-Portfolios versus CAPM

	Test Data (2016 - 2023)											
	AJ-GAN				AJ-Φ_ξ				AJ$^{(\text{bond})}$			
Decile	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p
1	7.9	0.8	0.12	0.90	2.7	-4.7	-1.36	0.17	5.6	-3.9	-0.96	0.34
2	10.2	-4.5	-0.89	0.38	6.5	-1.9	-0.63	0.53	9.8	-2.2	-0.57	0.57
3	15.8	-3.0	-0.56	0.57	9.2	-2.4	-0.74	0.46	14.3	-1.7	-0.45	0.65
4	14.2	-1.7	-0.40	0.69	9.8	-5.8	-1.80	0.07*	18.0	0.5	0.13	0.90
5	19.0	6.8	1.54	0.12	15.8	-0.6	-0.16	0.87	22.0	1.9	0.41	0.68
6	20.8	3.0	0.68	0.50	19.1	0.6	0.13	0.90	20.1	0.8	0.16	0.87
7	23.0	5.0	1.21	0.23	23.5	8.9	2.35	0.02**	24.3	5.6	1.06	0.29
8	25.6	6.8	1.74	0.08*	27.9	8.1	1.66	0.10*	32.2	11.0	1.82	0.07*
9	26.9	9.6	2.22	0.03**	31.8	16.4	2.94	0.00***	29.2	11.6	1.84	0.07*
10	29.3	6.1	1.41	0.16	36.9	24.0	3.87	0.00***	37.3	17.8	2.68	0.01***
10-1	21.4	5.4	0.92	0.36	34.2	28.7	4.42	0.00***	31.7	21.7	3.40	0.00***
GRS AP Test			χ^2_{10}	p			χ^2_{10}	p			χ^2_{10}	p
			21.4	0.018**			56.0	0.000***			21.7	0.017**

Notes. This table reports the results of regressions on the return of decile-sorted β-portfolios on the CAPM return factor. The loadings β are those produced by the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN model with the martingale penalty term (AJ-Φ_ξ), and the AJ-GAN model where the temporary component is built from a portfolio of bond indices (AJ$^{(\text{bond})}$). Portfolios are equally-weighted within each decile. The CAPM return factor is defined as the return on the market-weighted portfolio of stocks and is rebalanced weekly. The sample estimate of the unconditional expected return for each portfolio is annualized and presented under the heading $\mathbb{E}[R^e_{t+1}]$. The annualized pricing error (α), its t-stat (t) and the two-tailed p-value (p) are calculated from a projection of portfolio returns onto the return of the market-weighted portfolio. Excess returns and pricing errors are expressed as percentage units. The GRS asset pricing (AP) test statistic is defined in Gibbons et al. (1989) and has the null hypothesis that the CAPM correctly prices all decile portfolios. Newey-West standard errors are used to correct for heteroskedasticity and autocorrelation (using $T^{1/4} \cdot 0.75 \approx 2$ lags). Significance levels are specified as follows: * $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$.

The pattern of the t-stats suggests that the middle deciles behave more like the market factor and that more extreme deciles are loading onto new and unique signals. This assertion is corroborated by the observation that all pricing errors of the 10-1 long-short portfolios are significant at the 1% level. The results of the GRS test show that the null hypothesis that the CAPM correctly prices all decile portfolios can be rejected at the 10% level for all models, the 5% level for the GAN and EN, and at the 1% level for the EN. Adopting the traditional 5% threshold in asset pricing, the answer to RQ 4 is that the CAPM does not span all decile-sorted portfolios generated by the GAN and EN. However, it may or may not correctly price all FFN portfolios. The results of the GRS test indicate that the GAN and EN are able to generate unique signals for Nordic stocks at the weekly frequency. Table III illustrates the same regression results and GRS tests for the AJ-GAN, AJ-Φ_ξ, and AJ$^{(\text{bond})}$

models. The null that the CAPM correctly prices all decile portfolios is rejected at the 5% level for all models. Interestingly, the AJ-GAN model is the only model with an insignificant 10-1 decile portfolio. Again, these results remain robust when the models are benchmarked against the Fama and French (1992) 3-factor model and the and the Fama and French (2015) 5-factor model, see Table A18 and Table A19

IV.C Variable Importance

Figure VI illustrates the out-of-sample variable importance (VI) assigned to the characteristics by the GAN, FFN, and EN. To answer RQ 5, the results suggest that the FFN and EN are heavily driven by factors related to market frictions or limits-to-arbitrage (LTA). This is in line with the criticism of Avramov et al. (2023). In particular, the EN assigns more than 25% variable importance to standardized unexplained variation (SUV), and 3 of the 4 most important variables are related to frictions. The l_1-penalty of the validated EN leads to variable selection strongly in favor of frictions. In particular, only half of the available characteristics were included in the optimal model; see Panel B of Figure A26. In addition, the effect of the optimal l_2-penalty is to greatly shrink the impact of non-friction characteristics; see panel A of Figure A26. Surprisingly, the EN finds that none of the intangible factors is important, all of these factors have been regularized away. This is in contrast to the GAN, which assigns a high variable importance to accruals (AC).

The variable importance of the GAN and FFN is smoother because they do not impose the same strict form of variable selection or shrinkage as the EN. What distinguishes the GAN from the FFN is the dependence on frictions. The FFN finds that 7 of the 12 most important characteristics are related to LTA. Specifically, the most important variable is total assets (AT). Out of the GAN, FFN, and EN, only the GAN assigns the highest variable importance to a non-friction characteristic, that of profit margin (PM). In general, the prevalence of high-VI frictions is lower for the GAN. The adversarial nature and robust control in the GAN estimation may allow the pricing kernel to be less prone to LTA signals. An economic interpretation of this result is that the adversary could be trying to exploit mispricings in characteristics unrelated to frictions. For example, the adversary might be building arbitrage portfolios based on PM, NOA, and AC. The GAN model would then update the pricing kernel to reduce the prevalence of these specific arbitrage portfolios accordingly. Such a response from the pricing kernel would show up as a higher VI metric on these particular characteristics. However, like the EN, the GAN also asserts that SUV is one of the most important sources of predictability within the data. A commonality for the FFN and EN is that market capitalization (LME) appears as an important factor in explaining the returns of

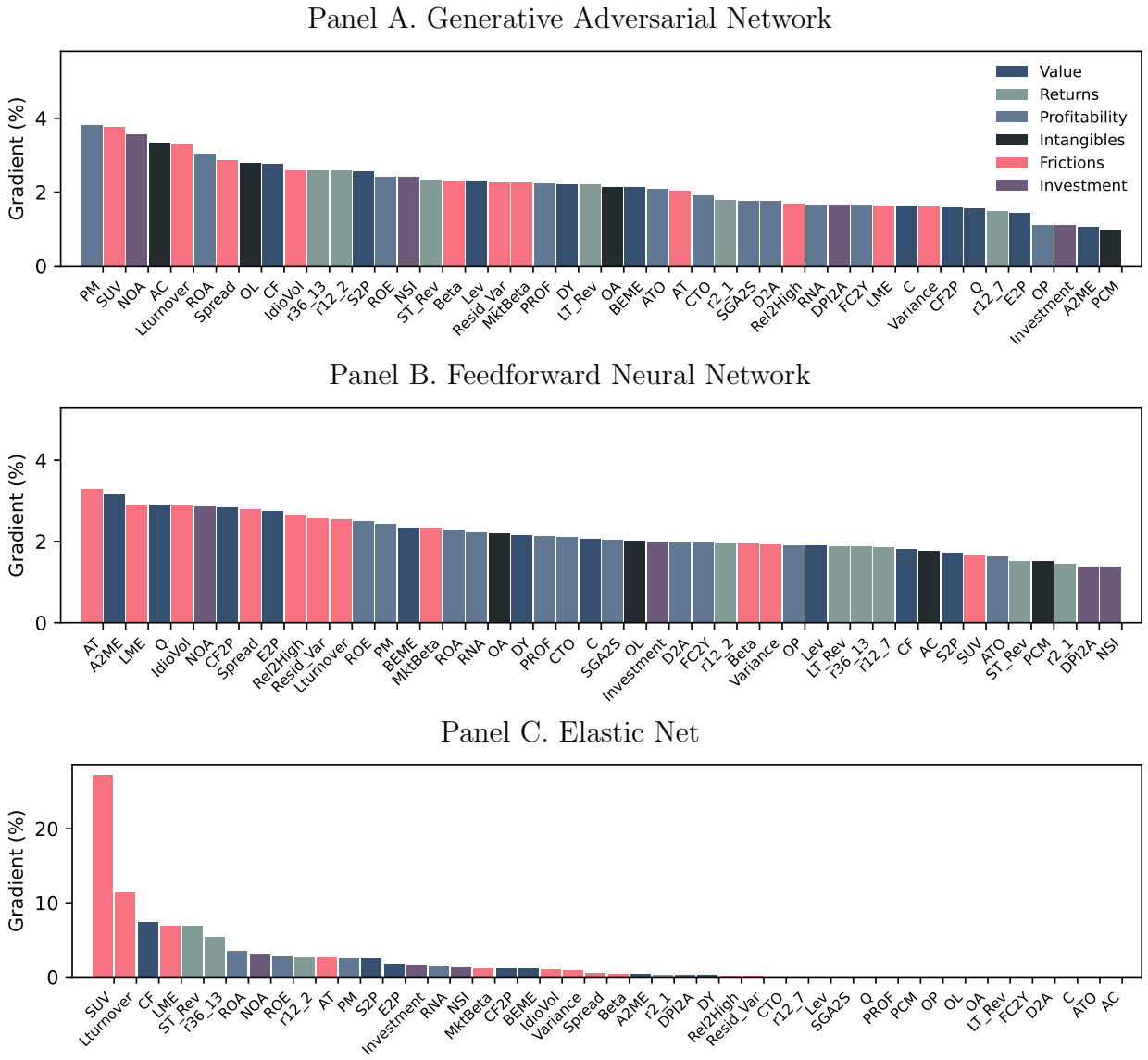


Figure VI

Variable Importance of Firm Characteristics

Notes. This figure provides answers to RQ 5 and RQ 6. The variable importance (VI) for each of the 46 firm characteristics is calculated as the normalized average absolute gradient; see Equation (24). During the calculation of the gradient for the GAN and FFN, macroeconomic variables are kept constant. The average absolute gradient for the EN simplifies to the absolute value of the coefficients $\hat{\theta}_{EN}$, and is also normalized to sum up to 1. All characteristics are colored according to their group category. All figures are derived from the test set.

Nordic stocks. This corroborates the importance of the quintessential size factor, which is well documented in the prior literature (Fama and French, 1992). According to the GAN, FFN, and EN models, the answer to RQ 6 is that standard unexplained variation (SUV), market capitalization (LME), profit margin (PM), and total assets (AT) are the most important characteristics for pricing Nordic stocks.

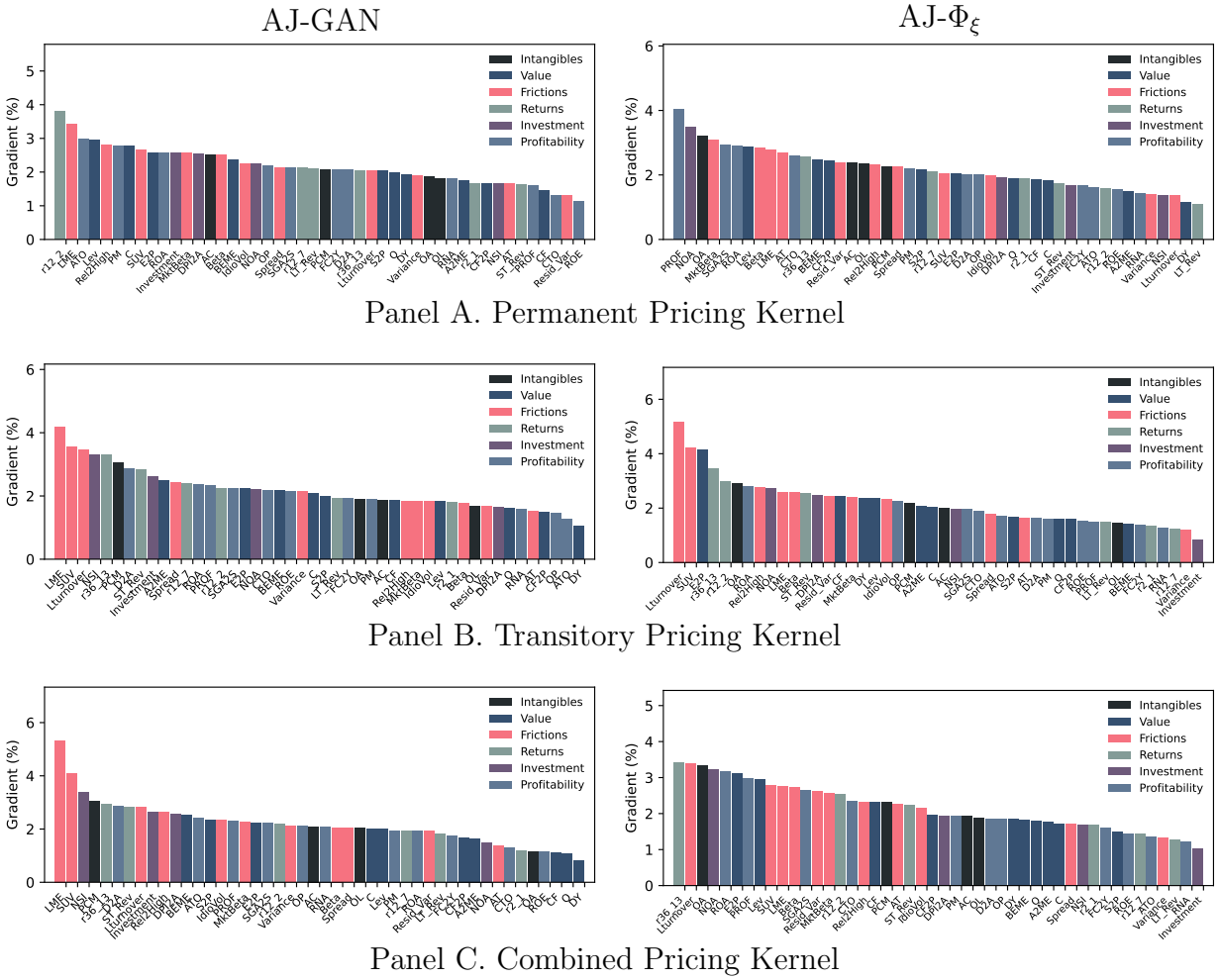


Figure VII

AJ Model Permanent-Transitory Decomposition of Characteristic Importance

Notes. This figure illustrates the assigned out-of-sample variable importance (VI) for the 46 firm characteristics of the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN) and the AJ-GAN with martingale regularization (AJ-Φ_ξ). The variable importance is defined as the normalized average absolute gradient. Panels A, B, and C correspond to the assigned VI of the permanent pricing kernel component $M_{t+1}^{(\mathrm{P})}$, transitory pricing kernel component $M_{t+1}^{(\mathrm{T})}$, and the combined pricing kernel $M_{t+1} = M_{t+1}^{(\mathrm{P})} \cdot M_{t+1}^{(\mathrm{T})}$, respectively. VI is normalized to sum up to 1. During the calculation of the gradient, the macroeconomic states are held constant.

Figure VII and Figure VIII illustrate the most important characteristics according to the transitory and permanent components of the extended GAN models. The transitory component of the AJ-GAN and the AJ-Φ_ξ models is primarily driven by turnover (Lturnover) and standard unexplained variation (SUV). In contrast, no characteristic related to market frictions is considered the most important characteristic by the permanent pricing kernel. Instead, momentum (r12_2) and profitability (PROF) are assigned the highest importance. In the combined pricing kernel, it is apparent that the AJ-GAN model assigns much higher

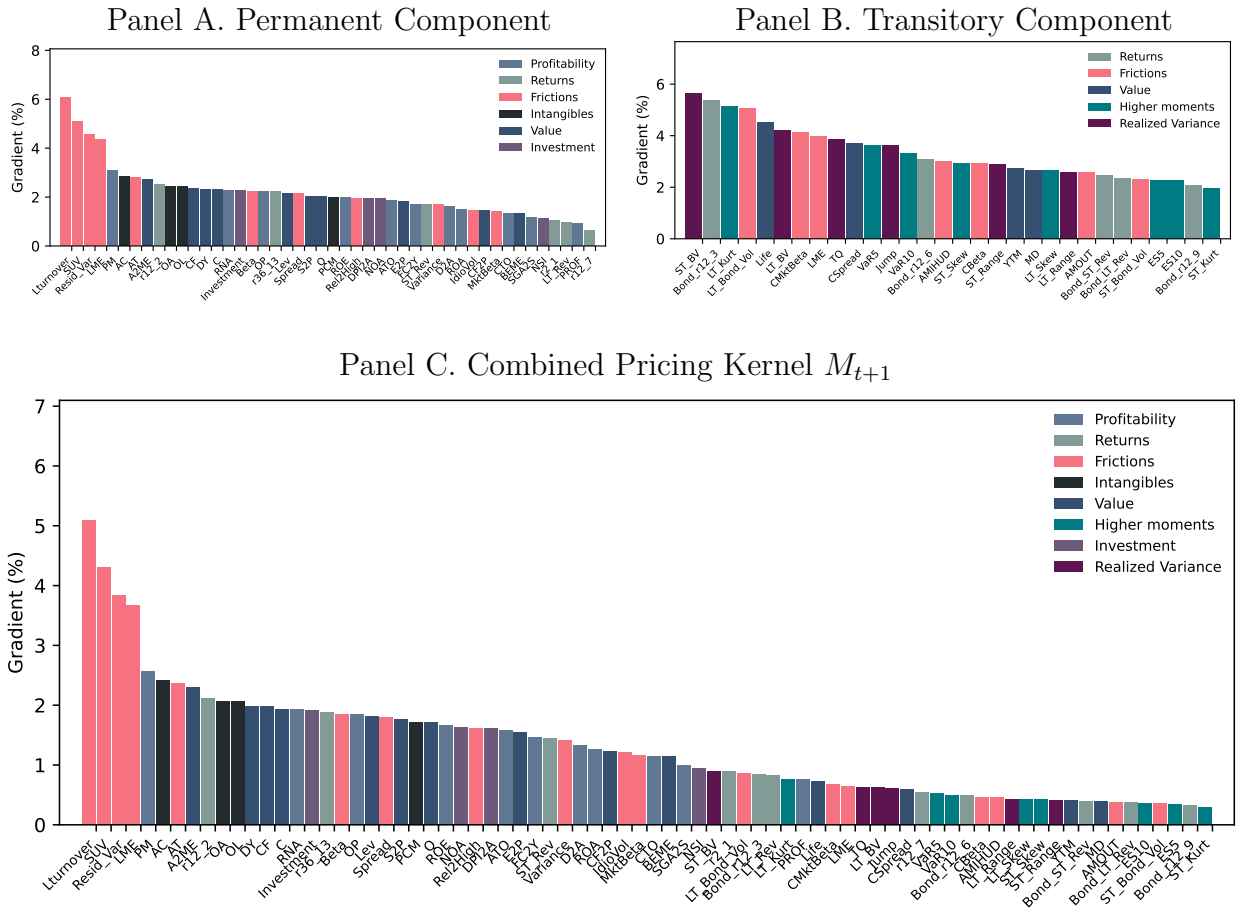


Figure VIII

Permanent-Transitory Decomposition of Characteristic Importance for the AJ$^{(\text{bond})}$ Model

Notes. This figure illustrates the assigned out-of-sample variable importance (VI) for the 46 firm characteristics and the 30 bond characteristics of the Alvarez and Jermann (2005) version of the generative adversarial network in which the transitory component is built using only bond data (AJ$^{(\text{bond})}$). The variable importance is defined as the normalized average absolute gradient. Panels A, B, and C correspond to the VI from the permanent component $M_{t+1}^{(\text{P})}$, the transitory component $M_{t+1}^{(\text{T})}$, and the combined pricing kernel $M_{t+1} = M_{t+1}^{(\text{P})} \cdot M_{t+1}^{(\text{T})}$, respectively. VI is normalized to sum up to 1. During the calculation of the gradient, all macroeconomic states are held constant at zero.

importance to frictions than the AJ-Φ_ξ model, which finds long-term momentum (r36_3) to be the most important characteristic. The economic rationale behind the martingale constraint might allows the model to reduce its dependence on frictions. In comparison, the permanent component of the AJ$^{(\text{bond})}$ model is highly skewed towards frictions. The Lturnover and the SUV are also among its most important firm characteristics. In contrast, the transitory component of the AJ$^{(\text{bond})}$ model depends only on bond characteristics. The most important of these is the short-term bipower variation (ST_BV), which is a realized variance metric. The combined pricing kernel of the AJ$^{(\text{bond})}$ model considers firm characteristics more important

than bond characteristics. Accordingly, the most important characteristics of the combined kernel are inherited directly from the permanent pricing kernel. To answer part (i) of RQ 9, it is apparent that at least one component of the pricing kernel is heavily driven by frictions, regardless of how the permanent-transitory decomposition is implemented.

IV.D Pricing Characteristic Portfolios

Table IV
Pricing Performance on Decile Sorted Characteristic Portfolios

	Time-Series R^2			Cross-Sectional R^2				Time-Series R^2			Cross-Sectional R^2		
Char.	GAN	FFN	EN	GAN	FFN	EN	Char.	GAN	FFN	EN	GAN	FFN	EN
Intangibles							Investment						
AC	0.71	0.25	0.30	0.79	0.48	0.34	Investment	0.70	0.31	0.29	0.95	0.65	0.50
OA	0.70	0.33	0.30	0.94	0.63	0.48	NOA	0.74	0.36	0.24	0.88	0.74	0.04
OL	0.70	0.35	0.33	0.86	0.55	0.46	NSI	0.69	0.34	0.31	0.81	0.68	0.41
PCM	0.66	0.28	0.31	0.91	0.51	0.61	ATO	0.69	0.34	0.33	0.91	0.64	0.58
Frictions							Profitability						
AT	0.79	0.37	0.21	0.91	0.87	-0.22	CTO	0.68	0.38	0.30	0.88	0.72	0.54
Beta	0.51	0.25	0.25	0.76	0.47	0.49	D2A	0.72	0.36	0.34	0.92	0.66	0.49
IdioVol	0.63	0.32	0.28	0.79	0.40	0.19	FC2Y	0.65	0.25	0.30	0.89	0.59	0.48
LME	0.81	0.46	0.04	0.97	0.60	0.02	OP	0.65	0.36	0.26	0.85	0.73	0.55
Lturnover	0.68	0.16	0.19	0.76	0.41	0.53	PM	0.67	0.25	0.30	0.83	0.62	0.37
MktBeta	0.49	0.25	0.25	0.71	0.51	0.58	PROF	0.66	0.31	0.27	0.95	0.74	0.60
Rel2High	0.64	0.38	0.32	0.89	0.68	0.44	RNA	0.68	0.36	0.30	0.90	0.59	0.48
Resid_Var	0.65	0.32	0.29	0.90	0.36	0.32	ROA	0.65	0.30	0.25	0.92	0.67	0.50
SUV	0.73	0.21	0.09	0.97	0.35	0.19	ROE	0.67	0.31	0.29	0.89	0.61	0.58
Spread	0.69	0.30	0.25	0.81	0.46	0.64	SGA2S	0.65	0.24	0.26	0.95	0.52	0.53
Variance	0.59	0.30	0.29	0.77	0.42	0.28	A2ME	0.69	0.40	0.34	0.91	0.77	0.40
Returns							Value						
LT_Rev	0.63	0.30	0.32	0.81	0.67	0.47	BEME	0.72	0.45	0.35	0.90	0.62	0.61
ST_Rev	0.69	0.37	0.28	0.88	0.56	0.34	C	0.70	0.35	0.31	0.94	0.59	0.58
r12_2	0.67	0.34	0.32	0.89	0.71	0.42	CF	0.68	0.31	0.26	0.91	0.69	0.51
r12_7	0.65	0.33	0.30	0.92	0.62	0.51	CF2P	0.66	0.36	0.28	0.86	0.62	0.41
r2_1	0.69	0.36	0.34	0.91	0.65	0.33	DY	0.66	0.29	0.31	0.93	0.67	0.35
r36_13	0.63	0.31	0.29	0.78	0.75	0.21	E2P	0.66	0.31	0.25	0.86	0.68	0.40
							Lev	0.67	0.31	0.31	0.93	0.65	0.49
							Q	0.69	0.34	0.34	0.95	0.62	0.58
							S2P	0.67	0.33	0.33	0.93	0.61	0.37

Notes. This table reports the out-of-sample (OOS) time series explained variation (TS-R^2) and cross-sectional explained variation (CS-R^2) on 460 characteristically managed portfolios. The metrics are presented for the generative adversarial network (GAN), feedforward network (FFN), and the elastic net (EN). Each characteristically managed portfolio is constructed by sorting stocks into 10 deciles for each of the 46 characteristics ("Char."). Portfolios are weighted by market value within decile and are rebalanced weekly. Firms with missing characteristics are excluded from any portfolio. These results provide an additional robustness check for part (ii) of RQ 3

Table IV presents the out-of-sample TS-R^2 and CS-R^2 when the GAN, FFN, and EN are asked to price 460 characteristically managed portfolios. These portfolios are sorted each week into deciles according to each characteristic $C_{t,i}$ and are value-weighted within the

decile. In most cases, the GAN explain twice as much variation in individual portfolio returns, as measured by TS-R^2. Going from individual returns to characteristically managed test assets diversifies away some idiosyncratic noise in the data. This helps the 3 models identify characteristic-related signals. Empirically, this appears as significant improvements in TS-R^2 and CS-R^2 compared to Table I. As predicted by theory, all models have a meaningful explanatory power in average returns, as measured by a high CS-R^2. Still, the GAN is better at explaining the expected returns of the portfolios, without fail. What differentiates the FFN and EN is a tendency of FFN to achieve a higher CS-R^2 than the EN. This ranking of TS-R^2 and CS-R^2 is in line with Table I, suggesting that the answer to part (ii) of RQ 3 is robust. The economic interpretation of this finding is that the GAN is much better at explaining not only individual stock returns and their expectations, but also aggregates of stocks. In its role as a portfolio-building arbitrageur, the adversarial COND network successfully improves the ability of the GAN pricing kernel to price various portfolios of stocks. Additionally, characteristically managed portfolios are commonly used to construct risk factors (size, value, quality, momentum, etc.). This suggests that the GAN is better at identifying the underlying factor structure inherent in Nordic stock returns. Because the optimal EN ignores half of the characteristics, it regularizes away many weak but informative signals associated with the omitted characteristics. This appears in the data as a lower TS-R^2 and CS-R^2 on the characteristic portfolios compared to the GAN and FFN.

A detailed account of the pricing performance of the GAN, FFN, EN on the 4 most important variables according to the GAN is provided in Table V. The FFN has a lower TS-R^2 in terms of profit margin, standard unexplained variation, and accrual because it prices the most extreme decile portfolios poorly. Apart from SUV, the EN has a much more stable TS-R^2 across deciles. In theory, the adversarial estimation of the GAN pricing kernel should minimize the worst pricing errors. This is also the case, as the pricing errors α of any individual decile are much lower in magnitude for the GAN compared to the FFN and EN. Because pricing errors translate directly to explanatory power in average returns, this is also why the GAN achieves a higher CS-R^2 in the end. This result is robust when considering other characteristics in detail. See Table A2 and Table A3 for the pricing performance on the 4 most important characteristics according to the FFN and EN, respectively.

Table VI illustrates the explanatory power of the AJ-GAN, AJ-Φ_ξ, and AJ$^{(\text{bond})}$ models on the 460 characteristic managed portfolios. When comparing these metrics with those of the baseline GAN in Table IV, it is apparent that all extended GAN models consistently price the portfolios better. This means that the incremental gain in explanatory power associated with the permanent-transitory constraint is robust. Of the extended GAN models, it is apparent that the AJ$^{(\text{bond})}$ model performs the best in terms of explaining individual portfolio returns

Table V

Pricing the Most Important Characteristic Portfolios (according to GAN)

	Profit Margin (PM)						Standard Unexplained Variation (SUV)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN
1	0.52	-0.01	0.24	-0.01	0.28	0.00	0.70	-0.23	0.57	0.11	0.48	0.22
2	0.50	0.25	0.27	0.29	0.18	0.35	0.71	-0.18	0.48	-0.03	0.25	0.20
3	0.70	0.36	0.29	-0.16	0.03	0.07	0.74	0.06	0.35	0.01	0.19	0.18
4	0.69	0.37	0.26	-0.09	0.01	0.14	0.79	0.14	0.27	0.12	0.42	0.34
5	0.76	0.17	0.41	0.13	0.24	0.28	0.74	0.18	0.17	0.00	0.24	0.25
6	0.77	0.09	0.40	0.15	0.37	0.32	0.77	0.33	0.01	0.00	0.26	0.30
7	0.76	0.29	0.36	-0.01	0.15	0.13	0.72	0.27	-0.10	-0.04	0.09	0.30
8	0.78	0.32	0.31	-0.13	0.02	0.15	0.75	0.45	-0.22	-0.06	0.04	0.25
9	0.71	0.42	0.20	0.02	0.13	0.36	0.72	0.53	-0.36	-0.05	0.09	0.32
10	0.66	0.36	0.27	0.02	0.19	0.37	0.71	0.61	-0.46	-0.01	0.06	0.40
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.67	0.25	0.30	0.83	0.62	0.37	0.73	0.21	0.09	0.97	0.35	0.19
	Net Operating Assets (NOA)						Accrual (AC)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN
1	0.57	0.37	0.34	-0.11	-0.03	0.07	0.57	0.42	0.26	0.20	0.27	0.16
2	0.71	0.56	0.13	0.19	0.34	0.53	0.70	0.51	0.28	-0.04	-0.06	0.18
3	0.73	0.39	0.16	0.02	0.20	0.34	0.70	0.44	0.36	-0.23	-0.20	-0.10
4	0.80	0.43	0.12	-0.17	-0.14	0.14	0.68	0.51	0.40	0.23	0.23	0.42
5	0.70	0.29	0.22	0.19	0.20	0.49	0.67	0.35	0.25	-0.09	0.07	0.13
6	0.80	0.03	0.24	-0.00	0.13	0.30	0.73	0.22	0.32	-0.03	0.12	0.21
7	0.82	0.33	0.33	-0.03	0.07	0.31	0.75	0.18	0.40	0.02	0.17	0.20
8	0.81	0.34	0.29	-0.08	0.08	0.21	0.76	0.10	0.38	0.16	0.32	0.43
9	0.86	0.33	0.41	-0.06	0.08	0.19	0.77	-0.12	0.05	-0.14	0.18	0.03
10	0.67	0.44	0.37	0.03	0.09	0.17	0.69	-0.11	0.31	0.02	0.41	0.36
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.74	0.36	0.24	0.88	0.74	0.04	0.71	0.25	0.30	0.79	0.48	0.34

Notes. This table records out-of-sample (OOS) metrics of time series explained variation (TS-R^2) and weekly pricing errors (α) on 40 characteristically managed portfolios. TS-R^2 and explained variation in average returns (CS-R^2) are presented across all deciles on row "All". Pricing errors are expressed in weekly percentage units. The metrics are presented for the generative adversarial network (GAN), feedforward network (FFN), and the elastic net (EN). The characteristically managed portfolios are constructed by sorting stocks into deciles for each characteristic. Portfolios are value-weighted and rebalanced weekly within decile. The characteristics presented here are the 4 most important characteristics as determined by the saliency analysis for the GAN model in Figure VI. These characteristics are profit margin (PM), standard unexplained variation (SUV), net operating assets (NOA), and accrual (AC).

almost across the board. In terms of explaining the average returns, AJ-GAN and AJ$^{(\text{bond})}$ alternate in which model produces the highest CS-R^2. This ranking of the AJ-GAN, AJ-Φ_ξ, and AJ$^{(\text{bond})}$ persists when considering pricing errors α and TS-R^2 of the individual characteristic managed portfolios. See Table A4, Table A5, and Table A6 for these metrics on the most important characteristics according to the extended GAN models.

Table VI
Extended GAN Model Pricing Performance on Decile Sorted Characteristic Portfolios

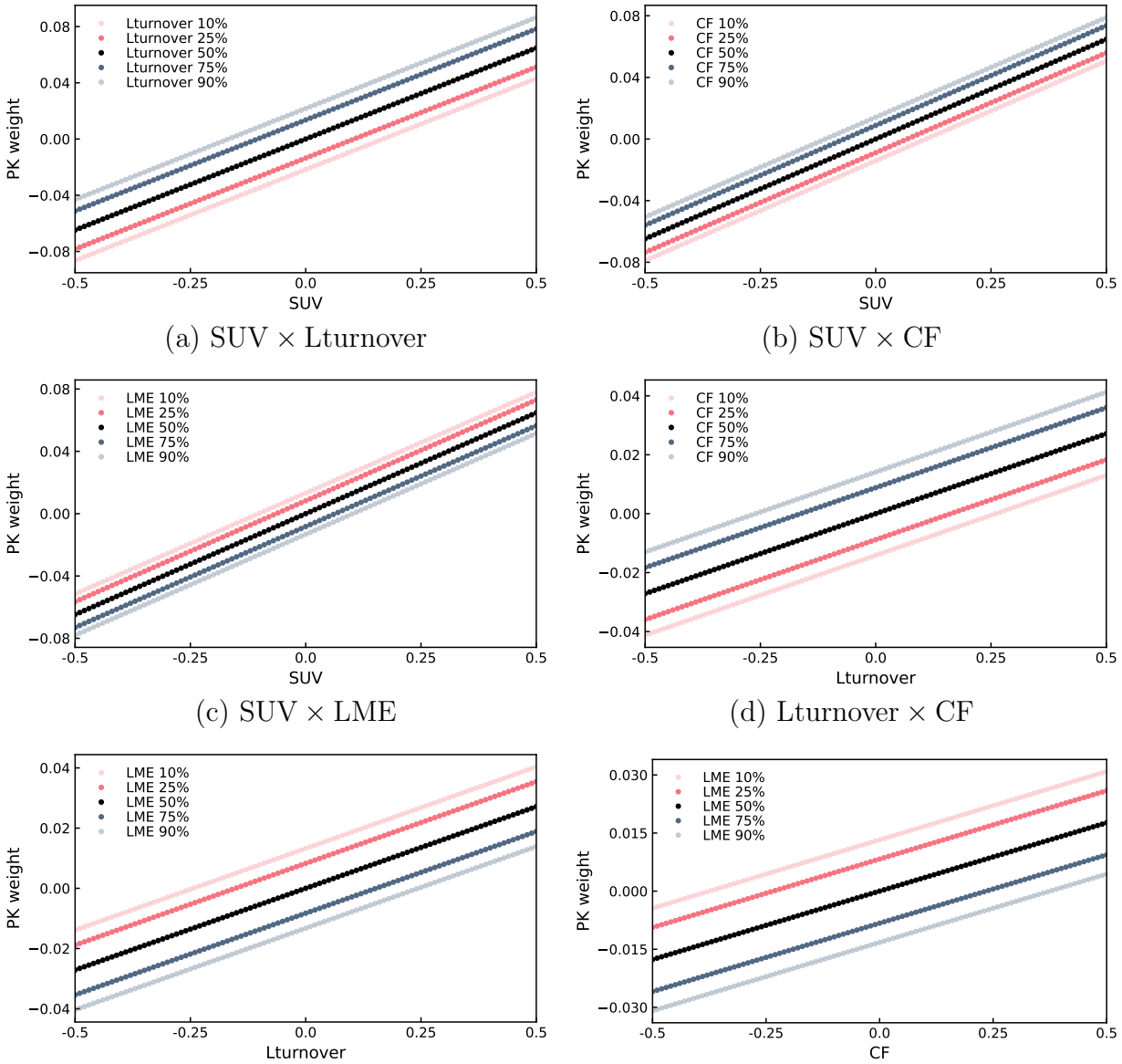
	Time-Series R^2			Cross-Sectional R^2				Time-Series R^2			Cross-Sectional R^2		
Char.	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	Char.	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$
Intangibles							Investment						
AC	0.73	0.73	0.73	0.82	0.83	0.84	Investment	0.72	0.72	0.73	0.96	0.97	0.96
OA	0.72	0.72	0.72	0.93	0.93	0.91	NOA	0.76	0.76	0.76	0.86	0.88	0.86
OL	0.73	0.72	0.73	0.87	0.88	0.87	NSI	0.73	0.73	0.74	0.85	0.85	0.86
PCM	0.69	0.69	0.70	0.93	0.92	0.90	ATO	0.71	0.71	0.71	0.89	0.90	0.88
Frictions							Profitability						
AT	0.79	0.80	0.80	0.89	0.91	0.89	CTO	0.71	0.70	0.71	0.89	0.89	0.88
Beta	0.63	0.60	0.66	0.91	0.89	0.92	D2A	0.74	0.74	0.74	0.93	0.93	0.93
IdioVol	0.69	0.69	0.71	0.84	0.80	0.83	FC2Y	0.66	0.66	0.67	0.94	0.95	0.96
LME	0.85	0.83	0.85	0.98	0.98	0.97	OP	0.69	0.68	0.70	0.86	0.85	0.84
Lturnover	0.74	0.73	0.75	0.87	0.87	0.89	PM	0.72	0.72	0.73	0.87	0.87	0.85
MktBeta	0.62	0.58	0.65	0.85	0.83	0.86	PROF	0.70	0.69	0.70	0.97	0.97	0.96
Rel2High	0.70	0.67	0.72	0.90	0.88	0.91	RNA	0.70	0.70	0.71	0.88	0.88	0.85
Resid_Var	0.70	0.70	0.72	0.89	0.86	0.86	ROA	0.70	0.70	0.71	0.94	0.94	0.93
SUV	0.76	0.75	0.76	0.95	0.95	0.95	ROE	0.71	0.71	0.71	0.91	0.91	0.90
Spread	0.78	0.75	0.79	0.92	0.90	0.93	SGA2S	0.68	0.68	0.69	0.93	0.93	0.91
Variance	0.69	0.68	0.72	0.75	0.76	0.71	A2ME	0.72	0.71	0.72	0.93	0.92	0.92
Returns							Value						
LT_Rev	0.66	0.66	0.67	0.87	0.89	0.89	BEME	0.75	0.75	0.76	0.93	0.93	0.92
ST_Rev	0.72	0.71	0.72	0.81	0.79	0.75	C	0.72	0.72	0.72	0.94	0.95	0.94
r12_2	0.71	0.69	0.72	0.90	0.86	0.90	CF	0.71	0.71	0.72	0.93	0.93	0.90
r12_7	0.70	0.69	0.71	0.91	0.91	0.91	CF2P	0.71	0.71	0.72	0.92	0.93	0.94
r2_1	0.72	0.71	0.73	0.88	0.84	0.87	DY	0.70	0.70	0.71	0.94	0.94	0.93
r36_13	0.68	0.67	0.70	0.84	0.86	0.85	E2P	0.70	0.70	0.71	0.91	0.92	0.93
							Lev	0.70	0.69	0.70	0.95	0.95	0.97
							Q	0.72	0.71	0.72	0.96	0.96	0.96
							S2P	0.70	0.69	0.70	0.94	0.94	0.94

Notes. This table reports the out-of-sample (OOS) time series explained variation (TS-R^2) and cross-sectional explained variation (CS-R^2) on 460 characteristically managed portfolios. The metrics are presented for the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN including martingale regularization (AJ-Φ), and the AJ-GAN in which the temporary component depends on bond data (AJ$^{(b)}$). Each characteristically managed portfolio is constructed by sorting stocks into 10 deciles for each of the 46 characteristics ("Char."). Portfolios are weighted by market value within decile and are rebalanced weekly. Firms with missing characteristics are excluded from any portfolio.

IV.E Interactions and Nonlinearities

As a linear factor model, the elastic net may capture interactions in variables but not any other nonlinearities. Figures IX and A10 illustrate the predicted pricing kernel ω_{EN} as a function of pairwise interactions of the 4 most important characteristics according to the EN, holding all other characteristics at median levels.[38] Interactions in the EN pricing kernel appear as linear shifts in the pricing kernel. The economic magnitude of the interactions is significant, the optimal position can even switch from a long position to a short position

[38]Interactions were studied for *all* characteristics but I illustrate only the results for the most important characteristics for the sake of brevity. The patterns are similar across all characteristics.



(a) SUV × Lturnover

(b) SUV × CF

(c) SUV × LME

(d) Lturnover × CF

(e) Lturnover × LME

(f) CF × LME

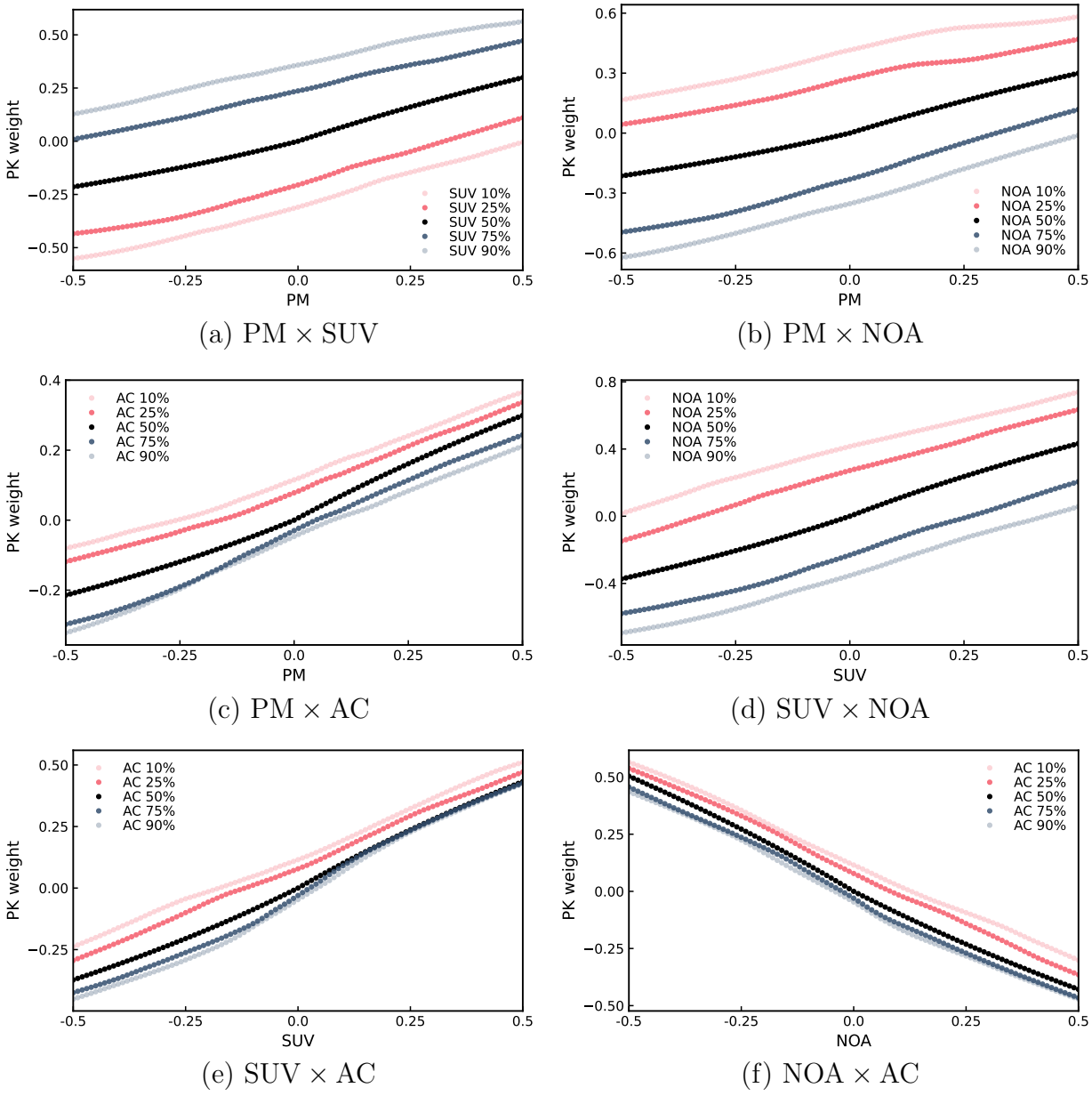
Figure IX

Characteristic Interactions within the Elastic Net Pricing Kernel $\hat{\omega}_{\text{EN}}$

Notes. The figure illustrates the out-of-sample interactions of predictors in the ensemble elastic net (EN) pricing kernel $\hat{\omega}_{\text{EN}}$ (before $\|\omega_t\|_1$ normalization). The characteristics used are the 4 most important characteristics according to Figure VI, i.e., standard unexplained variation (SUV), turnover (Lturnover), free cash flow to equity (CF), and market capitalization (LME). Interacted variables are set to the 10th, 25th, 50th, 75th, and 90th quantiles. All other characteristics are kept constant at median levels.

(and vice versa) when the main variable is near the median but the interacted variable changes. This underlines the importance of considering interactions in addition to standalone predictors in asset pricing. These results coupled with the high Sharpe ratio of the EN suggest that linear factor models work just fine, and ignoring complex nonlinearities is justified as

long as interactions are included in the specification.



(a) PM × SUV (b) PM × NOA

(c) PM × AC (d) SUV × NOA

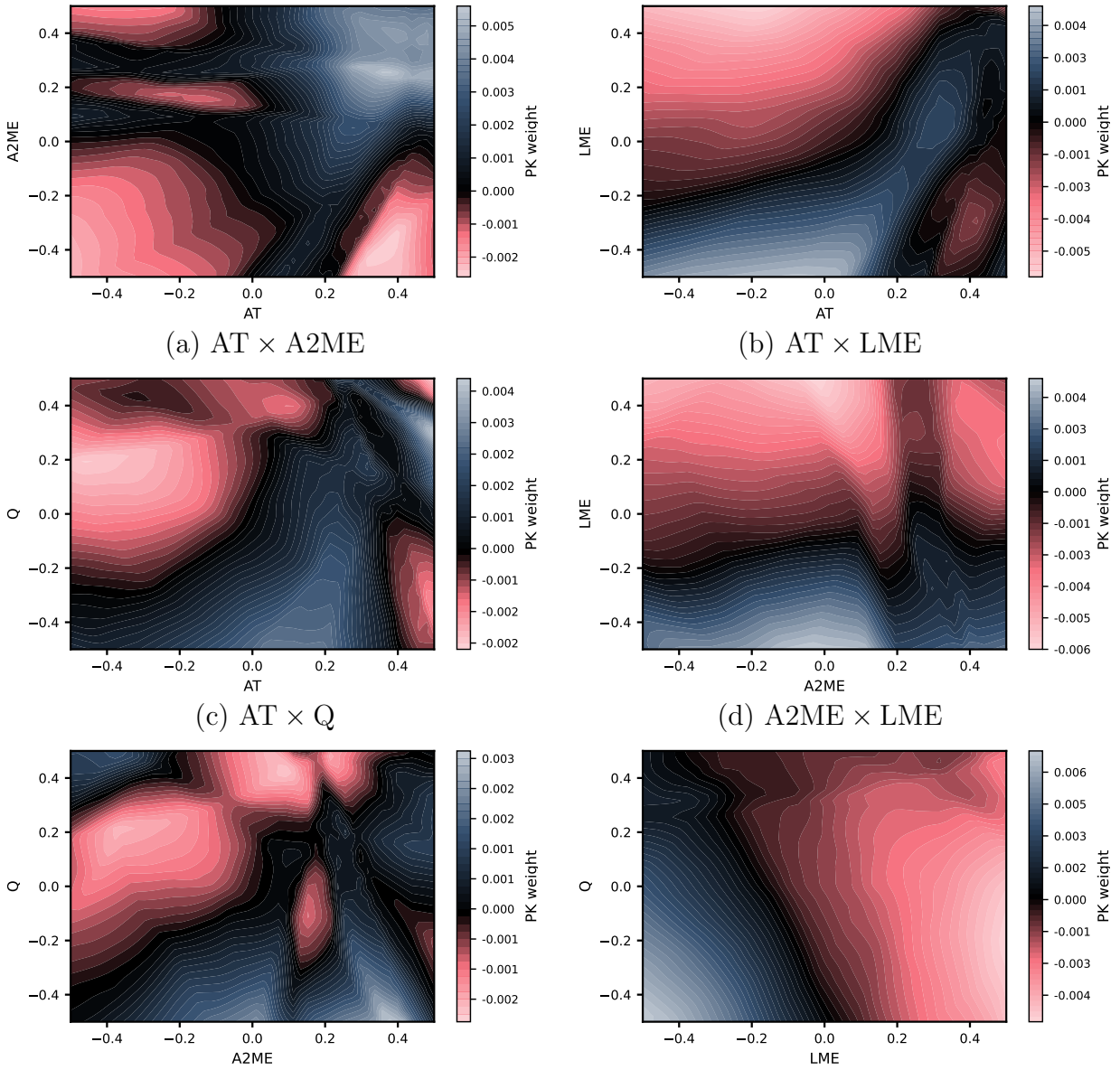
(e) SUV × AC (f) NOA × AC

Figure X

Characteristic Interactions within the GAN Pricing Kernel $\hat{\omega}_{\text{GAN}}$

Notes. The figure illustrates the out-of-sample interactions of predictors in the ensemble generative adversarial network (GAN) pricing kernel $\hat{\omega}_{\text{GAN}}$ (before $\|\omega_t\|_1$ normalization). The characteristics shown are the four most important characteristics according to the saliency analysis in Figure VI, i.e., profit margin (PM), standard unexplained variation (SUV), net operating assets (NOA), and accruals (AC). The interacted variables are set to the 10th, 25th, 50th, 75th, and 90th quantiles. All other firm characteristics are kept constant at median levels and hidden states are set to zero.

The GAN has the flexibility to capture complex non-linear patterns, but is still approximately linear in predictors; see Figure X. Like the EN, the patterns in the pricing kernel



(a) AT × A2ME (b) AT × LME

(c) AT × Q (d) A2ME × LME

(e) A2ME × Q (f) LME × Q

Figure XI

FFN Weights ω_{FFN} as a Function of Two Characteristics

Notes. The figure illustrates the out-of-sample nonlinear effect on the feedforward neural network (FFN) pricing kernel $\hat{\omega}_{\text{FFN}}$ (before $\|\omega_t\|_1$ normalization). Blue regions indicate where the model takes a long position and the red regions indicate short positions. The interactions considered are the 4 most important firm characteristics from Figure VI, i.e. total assets (AT), assets to market capitalization (A2ME), market capitalization (LME), and Tobin's Q (Q). Other firm characteristics are kept constant at median levels.

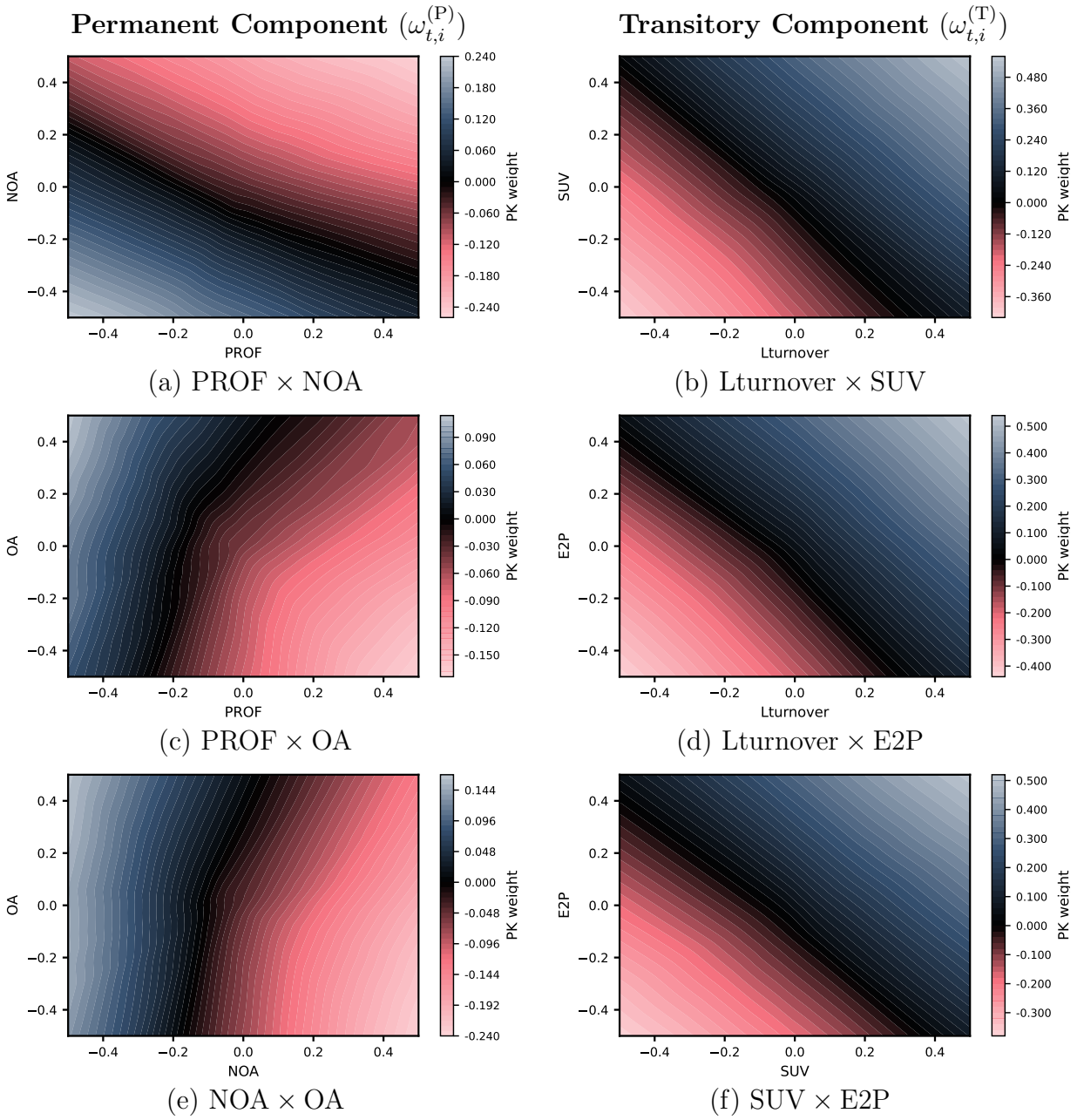
are monotonic and interactions matter as there are significant shifts in ω_{GAN} when the interacted variable changes. Apart from interactions, the GAN does not suggest that other complex nonlinearities matter for the pricing kernel. Although there is some slight evidence of convexity; see Figure A11, this small curvature could also be attributed to noise in the

estimation procedure. In comparison, pairwise interactions within the FFN pricing kernel ω_{FFN} paint a much less diffuse and more nonlinear picture; see Figure XI. Shifts in the landscape of the pricing kernel is highly localized. For example, the expected return may even change sign back-and-forth for small changes in the predictors. These extreme predictions allow the model to produce a Sharpe ratio that is higher than the market weighted portfolio of stocks. However, the instability of the FFN predictions explains why the GAN and EN produce stronger signals on out-of-sample data. The fact that the FFN pricing kernel produces these nonlinear patterns also has implications for the GAN model. Because the same ensemble learning methodology is used for all models, the smooth patterns within the GAN pricing kernel cannot simply be attributed to ensemble learning. Instead, the patterns that the GAN capture are robust as there is less variance across model fits. Because the GAN and EN are better at separating the signal from the noise, the results suggest that capturing interactions is of first-order importance. In comparison, capturing other nonlinearities, such as convexity, is only of second-order importance. To position these results within the debate around nonlinearities, these results agree with Chen et al. (2023a). To specifically answer RQ 7; interactions in predictors play an important role in pricing Nordic stocks, but other complex nonlinearities are less important.

See Figure XII, Figure A12, and Figure A13 for pairwise interactions of the most important variables for the extended GAN models. The results are the same as for the baseline GAN model, all extended GAN models have pricing kernels that are approximately linear in predictors, with some slight hints of convexity. Crucially, there are no visible differences in the patterns between the component kernels. This includes the transitory component of the $\text{AJ}^{(\text{bond})}$ model, which uses only bond characteristics. To answer part (ii) of RQ 9; even when you impose the permanent-transitory constraint, both the transitory and permanent components capture similar nonlinear patterns as the baseline GAN model. That is, the components depend heavily on interactions between characteristics, and apart from slight convex patterns there is no apparent evidence that higher order complexities play a big role.

V. Discussion

Research question RQ 1 asks which of three prominent machine learning models performs best in separating the signal from the noise in the new and lower signal-to-noise environment; the GAN of Chen et al. (2023a), the FFN of Gu et al. (2020), or the EN of Kozak et al. (2020). The results suggest that this is the GAN model, which generates a pricing kernel with an out-of-sample Sharpe ratio of 2.41. This is almost 5 times higher than the historical estimate of 0.5 for US equities (Cochrane, 2005). 3 key factors contribute to this result.



(a) PROF × NOA (b) Lturnover × SUV

(c) PROF × OA (d) Lturnover × E2P

(e) NOA × OA (f) SUV × E2P

Figure XII

AJ-Φ_ξ Weights $\omega_{\text{AJ}-\Phi}$ as a Function of Two Characteristics

Notes. The figure illustrates the out-of-sample nonlinear effect on the Alvarez and Jermann (2005) version of the generative adversarial network including martingale regularization (AJ-Φ_ξ) pricing kernel components $\hat{\omega}_{\text{AJ}-\Phi}^{(T)}$ and $\hat{\omega}_{\text{AJ}-\Phi}^{(P)}$ (before $\|\omega_t\|_1$ normalization). The blue regions indicate where the model takes a long position and the red regions indicate short positions. These effects are illustrated pairwise for the 3 most important characteristics of the permanent component $\hat{\omega}_{\text{AJ}-\Phi}^{(P)}$ (left) and the transitory component $\hat{\omega}_{\text{AJ}-\Phi}^{(T)}$ (right). Variable importance is based on the saliency analysis in Figure VII. For the permanent component, these characteristics are profitability (PROF), net operating assets (NOA), and operating assets (OA). For the transitory component, these characteristics are turnover (Lturnover), standardized unexplained variation (SUV), and earnings to price (E2P). All other firm characteristics are held constant at median levels and hidden states are fixed at zero.

First, the GAN model directly measures the state of the business cycle, which is ignored by the FFN and EN models. By capturing macroeconomic states, the GAN is better positioned to price any implicit state-contingent claims that are inherent in Nordic stocks. Second, the no-arbitrage constraint in the GAN estimation anchors the model directly to asset pricing theory. This makes the model more customized to the domain-specific setting of asset pricing, allowing the model to generate a more informed pricing kernel. Third, adversarial estimation of the pricing kernel provides robust control in the sense of Hansen and Sargent (2001). Hypothetically, the unconditional fit of the GAN model could lead to a pricing kernel that proxies for the market-weighted portfolio of stock returns. This would be the CAPM pricing kernel, which prices stocks reasonably well. Economically, the adversary could simply build size and value sorted portfolios during the conditional estimation phase. These factors are commonly mispriced by the CAPM. Crucially, training the adversary will change the entire loss landscape of the objective function for the PK network. In the next step, the pricing kernel must respond to the adversary by pricing these arbitrage portfolios. It could do so by building a pricing kernel similar to the Fama and French (1992) 3-factor model, which includes size and value as risk factors. The adversary could then build momentum portfolios in line with Jegadeesh and Titman (1993), which are usually mispriced by the FF3 model. This iteration will again change the entire loss landscape for the pricing kernel. In the following pass, the pricing kernel might update itself to something that resembles the Fama-French-Carhart 4-factor model to price these new momentum portfolios. This hypothetical example underlines the way that robust control appears in the estimation. *Robust control makes the loss landscape dynamic rather than static.* In contrast, the FFN and EN operate only with static loss landscapes throughout the entire estimation procedure. Econometrically, this makes the FFN and EN more susceptible to local minima and overfitting. Economically, this allows the GAN to produce a more robust pricing kernel that is better at separating the signal from the noise out-of-sample.

Part (i) of RQ 8 asks if it is possible to improve the signal-to-noise ratio of the GAN model by including the permanent-transitory decomposition as an economic constraint on the pricing kernel. Theoretically, assuming that the baseline GAN model is able to find a pricing kernel that is already on the Hansen and Jagannathan (1991) variance bound, simply decomposing that pricing kernel will not alter the HJ bound on its own. As a result, we would expect the maximum attainable signal-to-noise ratio to remain unchanged after imposing the economic constraint. However, by decomposing the pricing kernel, each component can now explain different dynamics of asset returns. Therefore, it is possible that the decomposition will lead to an increase in explained variation of returns while maintaining a signal-to-noise ratio similar to the baseline model. Empirically, this is approximately the case for the

naive AJ-GAN and AJ-Φ_ξ models. Both models maintain a signal-to-noise ratio in line with the baseline GAN but see an economically meaningful uptick in explanatory power in terms of individual returns, average returns, and portfolio returns. Therefore, in response to part (ii) of RQ 8, there are significant improvements to be made in explanatory power by imposing the permanent-transitory constraint. This level shift is not a fluke; it persists when considering 460 characteristically managed portfolios as test assets. This corroborates the principle established by Nagel (2021) and Avramov et al. (2023), that is, "machine learning + economic constraints" will allow the machine learning model to perform better at asset pricing tasks compared to uninformed machine learning models. I find that this also holds within the extended GAN models. The least informed model, the AJ-GAN, naively imposes the permanent-transitory decomposition without properly identifying either component. As a result, its pricing kernel sees a slight deterioration in signal-to-noise ratio compared to the other extended GAN models. When the economically motivated martingale constraint is imposed, the AJ-Φ_ξ model can now outperform the baseline in terms of Sharpe ratio. When I identify the transitory pricing kernel using bond data in the AJ$^{(\text{bond})}$ model, the pricing kernel now has an out-of-sample Sharpe ratio of 2.85 compared to 2.41 of the baseline. This economically significant improvement suggests that the baseline pricing kernel might not be on the HJ bound after all, but further research should aim to confirm the statistical significance of these shifts.

RQ 2 calls to study whether the machine learning models outperform conventional asset pricing models in terms of the Sharpe ratio in the weekly Nordic setting. The answer is important for international investors and short-term traders looking to decide whether it is worth investing the time and capital required to switch from passive investing to these computationally intensive machine learning models. I find that all machine learning models generate economically and statistically significant gains in Sharpe ratio against not only the benchmark CAPM, but also the Fama-French 3-factor and 5-factor models. RQ 3 asks how strong the predictive and explanatory power are for the GAN, FFN, and EN models. Overall the predictive relationship holds well for all 3 models out-of-sample. The GAN is able to generate a pricing kernel and risk loadings that match the canonical security market line particularly well. The FFN and EN have a slight bias in their intercepts but produce an otherwise satisfactory linear relationship. Using the SMLs, market practitioners can more accurately estimate the systematic risk they are exposed to, and hence make more informed inferences on the future returns they can expect. While this analysis considers compensation for systematic risk, an analysis of higher-order moments of the return distribution is omitted. It could be the case that the predictions of the GAN model is more prone to asymmetries or susceptible to tail events. Researchers or practitioners of quantitative risk management

should further study the behavior of value-at-risk (VaR), expected shortfall (ES), and other risk metrics of these models before using them for risk management purposes in practice.

All models have some form of explanatory power in average returns, as measured by cross-sectional R^2. However, using the time series R^2, only the GAN and its extensions have meaningful explanatory power in individual returns. The strong sparsity imposed by the EN model means that it captures only systematic factors that affect all assets. Because individual returns are heavily influenced by idiosyncratic noise, this results in an out-of-sample TS-R^2 close to zero, whereas the FFN and GAN are able to explain 3% and 10% of individual returns, respectively. A caveat of the sample periods used in this study is that the training period covers the very noisy period of 2000-2011. The models are trained over the dot-com bubble, the great financial crisis, and the start of the Eurozone debt crisis, a period in which the market portfolio only had a Sharpe ratio of 0.05. In such an environment, the static estimation procedure for the FFN and EN could lead to low TS-R^2 because it was extremely difficult to explain the noise in individual returns for this particular period, especially at a weekly frequency. Future research should consider different slices of the sample period and extend the study period further back in time to see if the results change with more history. Analyzing the dynamics of these machine learning pricing kernels on additional frequencies, such as hourly or even minute-by-minute data, is an interesting exercise. It is possible that these models can be employed by market microstructure researchers to provide a high-frequency anatomy of the pricing kernel, or to enhance our understanding of the behavior of microstructure noise.

Using the GRS test, the answer to RQ 4 is yes. Apart from the FFN, the machine learning models generate unique trading signals that are not spanned by the market-weighted portfolio of stocks, the FF3, or the FF5 factor models at a 5% significance level. If the signals are so strong and unique, why has the asset management industry not been able to capture these signals already? A possible explanation is related to the classical Roll (1997) critique. The only prediction of the CAPM is that the market portfolio is mean-variance efficient. Therefore, "testing the CAPM" using the market-weighted portfolio of stocks as a proxy is subject to the joint hypothesis problem. I cannot reject the CAPM, I can only say that the market proxy I used was inappropriate. Therefore, extended asset pricing tests might find that, in fact, the machine learning signals presented here are not unique. They could simply be signals generated from a higher-dimensional market model that includes additional risk factors and other asset classes such as real estate, private equity, and human capital, and so on. More advanced market proxies are left to future research. Another explanation is related to research question RQ 5. The findings suggest that the performance of the machine learning models is heavily influenced by market frictions and limits-to-arbitrage (LTA), especially the

FFN and EN models. While the baseline GAN model appear to be slightly is less dependent on frictions, the answer to part (i) of RQ 9 suggest that at least one of its components is driven by LTA. This is evidenced by the saliency analysis for the extended GAN models. As noted in Avramov et al. (2023), the performance of ML models can deteriorate in practice in the presence of frictions. Future researcher might find it reasonable to filter out stocks below a certain market size threshold. Although I do this implicitly by excluding all stocks on the small Nordic growth markets, more rigorous filters might be necessary to mitigate the impact of illiquidity and LTA. However, researchers should make sure that the sample is not too restrictive as a result of the filters, as it may introduce selection and survivorship bias. In this paper, the impact of trading costs, short-selling constraints, and high turnover are not adjusted for. The purpose of these asset pricing models are to separate the signal from the noise. The trading strategy only comes about as a by-product. Therefore, practical considerations of trading strategies are outside the scope of this paper. However, I make a few suggestions for practitioners. While l_1-normalization of portfolio weights mitigates outlandish long or short positions, market practitioners should build a customized position-sizing algorithm based on their own needs and preferences. For example, they could make the objective function more informed by penalizing large short positions to reduce the impact of short-constraints. Furthermore, high turnover can lead to large transaction costs, which may consume all profits. Practitioners may therefore consider updating portfolio weights in practice only when the expected gain of updating the weight outweighs the transaction costs. Alternatively, practitioners could include a penalty term that proxies for transaction costs. Such a multi-objective loss function would force the models to balance the gains of updating portfolio weights versus the transaction costs incurred.

In practice, it is reasonable to assume that there may exist a group of mispriced assets. Such assets will appear in the data with a high degree of return predictability and minimal risk (Kelly and Xiu, 2023). In theory, traders in a perfect and competitive market would have taken advantage of these mispricings. The issue is that markets are not frictionless in practice. With LTA, such as trading costs, short-constraints, illiquidity, bid-ask spreads, slippage, etc., traders cannot act on these predictabilities and leave some mispricings untouched. A data mining researcher might look at the data and conclude that there are tons of predictable patterns and pervasive mispricings. This "on-paper" predictability appears only to rationalize the LTA present in the market. This is the least economically interesting type of predictability; researchers are interested in predictable patterns due to systematic risk exposure. If the econometrician builds a pricing kernel using such data, the pricing kernel will appear to have a very high signal-to-noise ratio simply because the model "shoe-horns" these LTA-related pseudopredictabilities together with economically meaningful predictabilities. It

is not unlikely that the machine learning models considered in this study conflate the effect of LTA with that of systematic risk, which translates into the appearance of a high signal-to-noise ratio. The saliency analysis points in this direction, which confounds any economic conclusions that can be drawn from the models. The econometrician wants a pricing kernel that equilibrates prices to their fair values absent any LTA (Kelly and Xiu, 2023). On the back of the envelope, if the market Sharpe ratio of 0.72 constitutes *all* available systematic risk, then this means that only about 30% of the 2.41 Sharpe ratio of the GAN model can be attributed to economically meaningful predictability. Although this argument takes one extreme, it illustrates that the clean pricing kernel is severely obfuscated by LTA. The econometrician is left with the daunting task of peeling off layers of individual LTA effects until she is able to isolate a clean and economically meaningful pricing kernel. For her to do so, theoretical advances have to be explored. To remove all frictions, we need a unified theory on how LTA impacts return predictability, both on a standalone basis and in unison. On the empirical side, combining the econometric toolbox with machine learning methods is likely going to be beneficial to this end. As shown in this paper, different model architectures, regularization methods, and objective functions can lead to very different results. Because there is no universally best machine learning model for all tasks, researchers should continue to explore which configurations can help to unearth the economically interesting pricing kernel.

To answer RQ 6, the GAN, FFN and EN find that the profit margin (PM), standard unexplained variation (SUV), market capitalization (LME), and total assets (AT) are the most important factors for the pricing of Nordic stocks. While the extended GAN models corroborate that LME and SUV are important, they also show a tilt towards turnover (Lturnover). Because market capitalization is closely related to the size factor of Fama and French (1992), this suggests that the ML pricing kernels successfully incorporate conventional risk factors into their estimation. In this paper, 46 firm characteristics that span 6 categories were considered. Future studies could incorporate additional predictors and additional categories. Furthermore, to isolate variable importance within category, future researchers can perform additional analyzes by sequentially restricting the subset of predictors to one group at a time. The answer to the research question RQ 7 confirms the findings of Chen et al. (2023a). That is, researchers and practitioners can make first-order improvements in their asset pricing models by capturing pairwise predictor interactions. However, capturing more complex nonlinearities, such as convexity, is likely to only yield second-order improvements in performance. To conclude with part (ii) of RQ 9, the extended GAN models are also approximately linear in pairwise predictors. Even within each component of the pricing kernel, capturing interactions is much more important than other complex nonlinearities. While it is possible that unobservable higher-dimensional nonlinear patterns may be present, the

methods by which the researcher can "break open the black box" are limited. Investigating 3-dimensional nonlinearities would still leave us to wonder if fourth-order nonlinearities are present, etc. Further advances in the machine learning literature on how to open the black box will allow researchers to extract more economic content from these asset pricing models.

VI. Concluding Remarks

The research questions studied in this paper are motivated by the apparent bias in the prior literature to evaluate machine learning models using only US data at a monthly frequency. By studying Nordic stock returns at a weekly frequency, I make my first contribution to the literature by showing that the performance of 3 prominent machine learning models translate well to this lower signal-to-noise environment. In particular, the GAN, FFN, and EN are much better at separating the signal from the noise than conventional asset pricing models such as the CAPM, FF3, and FF5 factor models. Among these machine learning models, the GAN stands out due to its high out-of-sample performance in Sharpe ratio, predictive power, and explanatory power. The crucial strength of the GAN model is its adversarial estimation procedure, the ability to estimate hidden macroeconomic states, and the inclusion of the no-arbitrage constraint. Still, the established notion that "machine learning + economic constraints = informed asset pricing model" can lead to even further incremental gains in the GAN model. This is the second contribution of this paper. With some asterisks along the way, the findings of this paper suggest that by imposing the permanent-transitory constraint in an economically informed way, further improvements in both the signal-to-noise ratio and explanatory power can be achieved in the GAN model. In particular, the AJ-Φ_ξ and AJ$^{(\text{bond})}$ models outperform the baseline GAN model in terms of out-of-sample Sharpe ratio and explanatory power. In contrast, using a naive implementation of the permanent-transitory constraint leads to a slight decrease in performance, as shown by the performance of the AJ-GAN model. This suggests that economic constraints cannot be added ad hoc, but have to be constructed in an informed way to produce robust economic gains. These results suggest that future researchers can profit by extending existing machine learning models for other economic constraints not already considered in this paper. Specifically, I showed how the original "1 versus 1" zero-sum game of the GAN can be extended to a game of "2 versus 1". With rapidly improving computational power, future researchers can estimate more general "N-versus-M" (non-zero sum) games. Similar to the naive implementation of the permanent-transitory constraint, such games may not lead to improved signal-to-noise ratios on their own. But by imposing additional economic constraints motivated by theoretical results from game theory, these pricing kernels can become better behaved and potentially

lead to improvements.

Although the findings suggest that machine learning models outperform conventional asset pricing models such as the CAPM, FF3, and FF5 factor models, the results also show that the signals generated by the ML models are primarily driven by market frictions and limits to arbitrage. Therefore, imposing economic constraints that will minimize the impact of LTA, illiquidity, and microstructure noise will become a necessary future extension. Such extensions, which would involve developing asset pricing-specific regularization methods, can lead to significant advances in identifying the LTA-free pricing kernel. TO this end, borrowing ideas from the informed machine learning literature should prove fruitful (Von Rueden et al., 2023). This goes for both the combined pricing kernel or its components, which are all obfuscated by layers of LTA. While I use a novel dataset in this paper, there is still ample room to contribute to the literature by exploring the performance of the models on (1) even higher frequency data, (2) other asset classes, or (3) in developing economies. Furthermore, the findings suggest that pairwise interactions are much more important than other nonlinearities, but future researchers should try to explore the impact of higher-order nonlinearities once the machine learning literature has figured out ways to extract such information in an interpretable way. Future researchers can also consider other model architectures for the GAN, in particular, by experimenting with other high-capacity RNN models such as transformers. By improving the estimates of the hidden macroeconomic states, researchers and practitioners can improve the ability of the model to price contingent claims. In general, the abundant flexibility of machine learning models means that there are many avenues for future researchers to make contributions. Regardless of the form of these extensions, the findings of this paper suggest that improvements are more likely to come from making machine learning models more informed by imposing economically meaningful constraints.

References

Alvarez, F. and Jermann, U. J. (2005). Using Asset Prices to Measure the Persistence of the Marginal Utility of Wealth. *Econometrica*, 73(6):1977–2016. doi: 10.1111/j.1468-0262.2005.00643.x.

Amihud, Y. (2002). Illiquidity and Stock Returns: Cross-Section and Time-Series Effects. *Journal of Financial Markets*, 5(1):31–56. doi: 10.1016/S1386-4181(01)00024-6.

Athey, S. (2019). *The Impact of Machine Learning on Economics*, pages 507–552. University of Chicago Press, Chicago.

Avramov, D., Cheng, S., and Metzker, L. (2023). Machine Learning vs. Economic Restrictions: Evidence from Stock Return Predictability. *Management Science*, 69(5):2587–2619. doi: 10.1287/mnsc.2022.4449.

Backus, D., Chernov, M., and Zin, S. (2014). Sources of Entropy in Representative Aget Models. *Journal of Finance*, 69(1):51–99. doi: 10.1111/jofi.12090.

Bai, J., Bali, T. G., and Wen, Q. (2019). Common Risk Factors in the Cross-Section of Corporate Bond Returns. *Journal of Financial Economics*, 131(3):619–642. doi: 10.1016/j.jfineco.2018.08.002.

Bakshi, G. and Ghabi-Yo, F. (2012). Variance bounds of the permanent and transitory components of stochastic discount factors. *Journal of Financial Economics*, 105(1):191–208. doi: 10.1016/j.jfineco.2012.01.003.

Bali, T. G., Subrahmanyam, A., and Wen, Q. (2021). Long-Term Reversals in the Corporate Bond Market. *Journal of Financial Economics*, 139(2):656–677. doi: 10.1016/j.jfineco.2020.08.007.

Bansal, R. and Yaron, A. (2004). Risks for the Long Run: A Potential Resolution of Asset Pricing Puzzles. *Journal of Finance*, 59(4):1481–1509. doi: 10.1111/j.1540-6261.2004.00670.x.

Barndorff-Nielsen, O. E. and Neil, S. (2004). Power and Bipower Variation with Stochastic Volatility and Jumps. *Journal of Financial Econometrics*, 2(1):1–37. doi: 10.1093/jjfinec/nbh001.

Barndorff-Nielsen, O. E. and Neil, S. (2006). Econometrics of Testing for Jumps in Financial Economics Using Bipower Variation. *Journal of Financial Econometrics*, 4(1):1–30. doi: 10.1093/jjfinec/nbi022.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning Long-Term Dependencies with Gradient Descent is Difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166. doi: 10.1109/72.279181.

Bianchi, D., Büchner, M., and Tamoni, A. (2021). Bond Risk Premiums with Machine

Learning. *Review of Financial Studies*, 34(2):1046–1089. doi: 10.1093/rfs/hhaa062.

Borovička, J., Hansen, L. P., Hendricks, and Scheinkman, J. A. (2011). Risk Price Dynamics. *Journal of Financial Econometrics*, 9(1):3–65. doi: 10.1093/jjfinec/nbq030.

Bryzgalova, S., Pelger, M., and Zhu, J. (2023). Forest through the trees: Building cross-sections of stock returns. *Journal of Finance*, forthcoming. doi: 10.2139/ssrn.3493458.

Camanho, N., Hau, H., and Rey, H. (2022). Global Portfolio Relabalcing and Exchange Rates. *Review of Financial Studies*, 35(11):5228–5274. doi: 10.1093/rfs/hhac023.

Campbell, J. Y. (2018). *Financial Decisions and Markets: A Course in Asset Pricing*. Princeton University Press, Princeton, NJ.

Campbell, J. Y., Lo, A. W., and MacKinlay, C. A. (1997). *The Econometrics of Financial Markets*. Princeton University Press, Princeton, NJ. doi: 10.2307/j.ctt7skm5.

Chen, L., Pelger, M., and Zhu, J. (2023a). Deep Learning in Asset Pricing. *Management Science*, 70(2):714–750. doi: 10.1287/mnsc.2023.4695.

Chen, L., Pelger, M., and Zhu, J. (2023b). Internet appendix for: Deep Learning in Asset Pricing. *Management Science*, 70(2):714–750. doi: 10.1287/mnsc.2023.4695.

Christensen, T. M. (2017). Nonparametric Stochastic Discount Factor Decomposition. *Econometrica*, 85(5):1501–1536. doi: 10.3982/ECTA11600.

Cochrane, J. H. (2005). *Asset Pricing: Revised Edition*. Princeton University Press, Princeton, NJ.

Cochrane, J. H. (2011). Presidential Adress: Discount Rates. *Journal of Finance*, 66(4):1047–1108. doi: 10.1111/j.1540-6261.2011.01671.x.

Cong, L. W., Tang, K., Wang, J., and Zhang, Y. (2020). AlphaPortfolio for Investment and Economically Interpretable AI. Working paper, Cornell University, Ithica, NY.

Cong, L. W., Tang, K., Wang, J., and Zhang, Y. (2021). Deep sequence modeling: Development and applications in asset pricing. *The Journal of Financial Data Science*, 3(1):28–42. doi: 10.3905/jfds.2020.1.053.

Dybvig, P. H. and Ingersoll Jr., J. E. (1982). Mean-Variance Theory in Complete Markets. *Journal o Business*, 55:233–251.

Epstein, L. G. and Zin, S. E. (1989). Substitution, Risk Aversion and the Temporal Behavior of Consumption and Asset Returns: A Theoretical Framework. *Econometrica*, 57(4):937–969. doi: 10.2307/1913778.

Erb, C. B., Harvey, C. R., and Viskanta, T. E. (1995). Country Risk and Global Equity Selection. *The Journal of Portfolio Management*, 1§(2):75–83. doi: 10.3905/jpm.1995.409504.

Fama, E. F. and French, K. R. (1992). The Cross-Section of Expected Stock Returns. *Journal*

of Finance, 47(2):427–465. doi: 10.1111/j.1540-6261.1992.tb04398.x.

Fama, E. F. and French, K. R. (1993). Common Risk Factors in the Returns on Stocks and Bonds. *Journal of Financial Economics*, 33(1):3–56. doi: 10.1016/0304-405X(93)90023-5.

Fama, E. F. and French, K. R. (2015). A Five-Factor Asset Pricing odel. *Journal of Financial Economics*, 116(1):1–22. doi: 10.1016/j.jfineco.2014.10.010.

French, K. (2024). Data library - description of fama/french 5 factors (2x3). `https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library/f-f_5_factors_2x3.html`. webpage.

Freyberger, J., Neuhierl, A., and Weber, M. (2020). Dissecting Characteristics Nonparametrically. *Review of Financial Studies*, 33(5):2326–2377. doi: 10.1093/rfs/hhz123.

Gebhardt, W. R., Hvidkjaer, S., and Swaminathan, B. (2005). The Cross-Section of Expected Corporate Bond Returns: Betas or Characteristics? *Journal of Financial Economics*, 75(1):85–114. doi: 10.1016/j.jfineco.2004.04.002.

Gibbons, M. R., Ross, S. A., and Shanken, J. (1989). A Test of The Efficiency of a Given Portfolio. *Econometrica*, 57(5):1121–1152. doi: 10.2307/1913625.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 27.

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *Review of Financial Studies*, 33(5):2223–2273. doi: 10.1093/rfs/hhaa009.

Gu, S., Kelly, B., and Xiu, D. (2021). Autoencoder Asset Pricing Models. *Journal of Econometrics*, 222(1):429–450. doi: 10.1016/j.jeconom.2020.07.009.

Hansen, L. and Jagannathan, R. (1997). Assessing Specification Errors in Stochastic Discount Factor Models. *Journal of Finance*, 52(2):557–90. doi: 10.2307/2329490.

Hansen, L. P. (1982). Large Sample Properties of Generalized Method of Moments Estimators. *Econometrica*, 50(4):1029–1054.

Hansen, L. P. (2012). Dynamic Valuation Decomposition within Stochastic Economies. *Econometrica*, 80(3):911–967. doi: 10.3982/ECTA8070.

Hansen, L. P. and Jagannathan, R. (1991). Restrictions on Intertemporal Marginal Rates of Substitutino Implied by Asset Returns. *Journal of Political Economy*, 99:226–262.

Hansen, L. P. and Richard, S. F. (1987). The Role of Conditioning Information in Deducing Testable Restrictions Implied by Dynamic Asset Pricing Models. *Econometrica*, 55(3):587–613. doi: 10.2307/1913601.

Hansen, L. P. and Sargent, T. J. (2001). Robust Control and Model Uncertainty. *American Economic review*, 91(2):60–66. doi: 10.1257/aer.91.2.60.

Hansen, L. P. and Scheinkman, J. A. (2009). Long-Term Risk: An Operator Approach. *Econometrica*, 77(1):177–234. doi: 10.3982/ECTA6761.

Hansen, L. P. and Scheinkman, J. A. (2012). Recursive utility in a markov environment with stochastic growth. *Proceedings of the National Academy of Sciences of the United States of America*, 109(30):11967–11972. doi: 10.1073/pnas.1200237109.

Heaton, J. B., Polson, N. G., and Witte, J. H. (2017). Deep Learning for Finance: Deep Portfolios. *Applied Stochastic Models in Business and Industry*, 33(1):3–12. doi: 10.1002/asmb.2209.

Hochreiter, J. (1991). Untersuchungen zu Dynamischen Neuronalen Netzen. Technical report, Diploma thesis, Institut für Informatik, Lehrstuhl Prof. Brauer, Technische Universität München.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780. doi: 10.1162/neco.1997.9.8.1735.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67. doi: 10.2307/1267351.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer Feedforward Networks are Universal Approximators. *Neural networks*, 2(5):359–366. doi: 10.1016/0893-6080(89)90020-8.

Houweling, P., Mentink, A., and Vorst, T. (2005). Comparing Possible Proxies of Corporate Bond Liquidity. *Journal of Banking & Finance*, 29(6):1331–1358.

Hull, I. (2021). *Machine Learning for Economics and Finance in TensorFlow 2*. Apress Berkeley, CA. doi: 10.1007/978-1-4842-6373-0.

Israel, R., Palhares, D., and Richardson, S. (2018). Common Factors in Corporate Bond Returns. *Journal of Investment Management*, 16(2):17–46. doi: 10.2139/ssrn.2576784.

Jegadeesh, N. and Titman, S. (1993). Returns to Buying Winners and Seeling Losers: Implications for Stock Market Efficiency. *Journal of Finance*, 48(1):65–91. doi: 10.2307/2328882.

Jostova, G., Nikolova, S. S., Philipov, A., and Stahel, C. W. (2013). Momentum in Corporate Bond Returns. *Review of Financial Studies*, 26(7):1649–1693. doi: 10.1093/rfs/hht022.

Kelly, B., Pruitt, S., and Su, Y. (2019). Characteristics are Covariances: A Unified Model of Risk and Return. *Journal of Financial Economics*, 134(3):501–524. doi:

10.1016/j.jfineco.2019.05.001.

Kelly, B. T. and Xiu, D. (2023). Financial Machine Learning. *Foundations and Trends in Finance*, 13(3):205–363. doi: 10.3386/w31502.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *Preprint*. doi: 10.48550/arXiv.1412.6980.

Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes. doi: 10.48550/arXiv.1312.6114.

Klambauer, G., Unterthiner, T., Mayr, A., and Hochreiter, S. (2017). Self-normalizing neural networks. *In Advances in Neural Information Processing Systems (NIPS)*. doi: 10.48550/arXiv.1706.02515.

Kleinberg, J., Ludwig, J., Mullainathan, S., and Obermeyer, Z. (2015). Prediction Policy Problems. *American Economic Review*, 105(5):491–95. doi: 10.1257/aer.p20151023.

Kozak, S., Nagel, S., and Santosh, S. (2020). Shrinking the cross-section. *Journal of Financial Economics*, 135(2):271–292. doi: 10.1016/j.jfineco.2019.06.008.

Lettau, M. and Pelger, M. (2020). Factors that fit the Time Series and Cross-Section of Stock Returns. *Review of Financial Studies*, 33(5):2274–2325. doi: 10.2139/ssrn.3211106.

Lewis, G. and Syrgkanis, V. (2018). Adversarial Generalized Method of Moments. Working paper, Stanford University, Stanford, NY. doi: 10.48550/arXiv.1803.07164.

Lo, A. W. (2019). *Adaptive Markets: Financial Evolution at the Speed of Thought*. Princeton University Press, Princeton, NJ. doi: 10.2307/j.ctvc77k3n.

Maggiori, M. (2017). Financial, Intermediation, International Risk Sharing, and Reserve Currencies. *American Economic Review*, 107(10):3038–3071. doi: 10.1257/aer.20130479.

Merton, R. C. (1980). On Estimating the Expected Return on the Market. *Journal of Financial Economics*, 8:323–361. doi: 10.3386/w0444.

Messmer, M. (2017). Deep Learning and the Cross-Section of Expected Returns. Working paper, University of St. Gallen, St. Gallen, Switzerland. doi: 10.2139/ssrn.3081555.

Moritz, B. and Zimmermann, T. (2016). Tree-Based Conditional Portfolio Sorts: The Relation between Past and Future Stock Returns. Working paper, University of Cologne, Cologne, Germany. doi: 10.2139/ssrn.2740751.

Nagel, S. (2021). *Machine Learning in Asset Pricing*. Princeton University Press, Princeton, NJ. doi: 10.23943/princeton/9780691218700.003.0001.

Nagel, S. and Singleton, K. (2011). Estimation and evaluation of conditional asset pricing models. *Journal of Finance*, 66(3):873–909. doi: 10.1111/j.1540-6261.2011.01654.x.

Nelder, J. A. and Mead, R. (1965). A Simplex Method for Function Minimization. *Computer*

Journal, 7(4):308–313. doi: 10.1093/comjnl/7.4.308.

Parkinson, M. (1980). The Extreme Value Method for Estimating the Variance of the Rate of Return. *The Journal of Business*, 53(1):61–65. doi: 10.1086/296071.

Qin, L. and Linetsky, V. (2016). Positive Eigenfunctions of Markovian Pricing Operators: Hansen-Scheinkman Factorization, Ross Recovery, and Long-Term Pricing. *Operations Research*, 64(1):99–117. doi: 10.1287/opre.2015.1449.

Qin, L. and Linetsky, V. (2017). Long-Term Risk: A Martingale Approach. *Econometrica*, 85(1):299–312. doi: 10.3982/ECTA13438.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). Why should i trust you? doi: 10.1145/2939672.2939778.

Roll, R. (1997). A Critique of the Asset Pricing Theory's Tests Part I: On Past and Potential Testability of the Theory. *Journal of Financial Economics*, 4(2):129–176. doi: 10.1016/0304-405X(77)90009-5.

Rossi, A. G. (2018). Predicting Stock Market Returns with Machine Learning. Working paper, University of Maryland, College Park, Maryland.

Schapire, R. E. (1989). The Strength of Weak Kearnability. In *30th Annual Symposium on Foundations of Computer Science*, pages 28–33. doi: 10.1109/SFCS.1989.63451.

Sharpe, W. (1964). Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk. *Journal of Finance*, 19:452–442. doi: 10.1111/j.1540-6261.1964.tb02865.x.

Singleton, K. J. (2005). *Empirical Dynamic Asset Pricing: Model Specification and Econometric Assessment*. Princeton University Press, Princeton, NJ.

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288. doi: 10.1111/j.2517-6161.1996.tb02080.x.

Varian, H. R. (2014). Big data: New tricks for econometrics. *Journal of Economic Perspectives*, 28(2):3–28. doi: 10.1257/jep.28.2.3.

Von Rueden, L., Mayer, S., Beckh, K., Georgiev, B., Giesselbach, S., Heese, R., Kirsch, B., Walczak, M., Pfrommer, J., Pick, A., Ramamurthy, R., Garcke, J., Bauckhage, C., and Schuecker, J. (2023). Informed Machine Learning - A Taxonomy and Survey of Integrating Prior Knowledge into Learning Systems. *IEEE Transactions on Knowledge and Data Engineering*, 53(1):614–633. doi: 10.1109/TKDE.2021.3079836.

Wolpert, D. H. and Macready, W. G. (1997). No Free Lunch Theorems for Optimization. *IEE Transactions on Evolutionary Computation*, 1(1):67–82. doi: 10.1109/4235.585893.

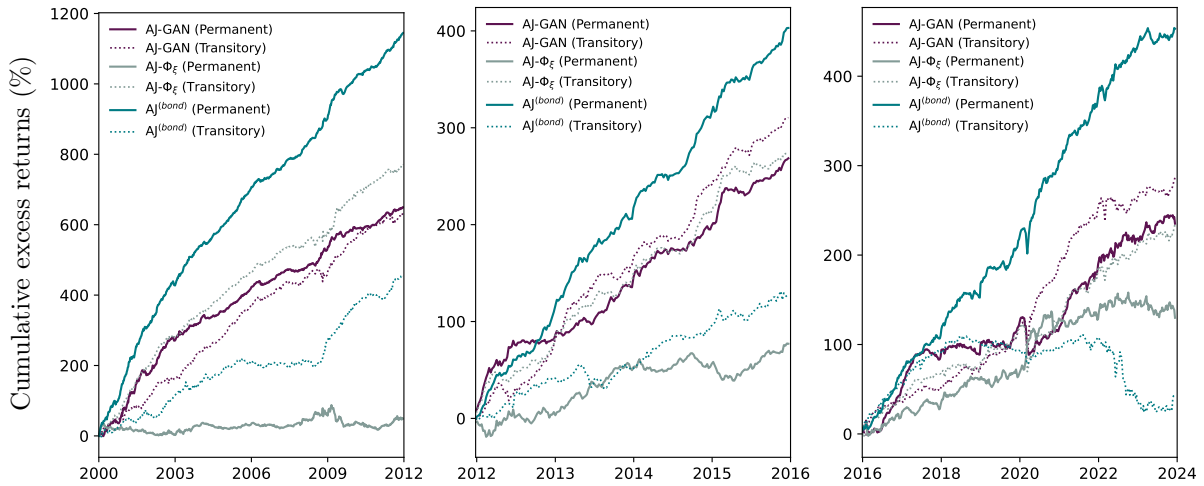
Zednik, C. (2021). Solving the Black Box Problem: A Normative Framework for Explainable Artificial Intelligence. *Philosophy & Technology*, 34:265–288. doi:

10.48550/arXiv.1903.04361.

Zou, H. and Hastie, T. (2005). Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 67(2):301–320. doi: 10.1111/j.1467-9868.2005.00503.x.

Appendix

A1. Supplementary Results and Hidden States



(a) Training data (b) Validation data (c) Testing data

Figure A1

Time Series of Permanent and Transitory Pricing Kernels

Notes. This figure illustrates the empirical cumulative excess returns of each component pricing kernel for the extended GAN models. These models include the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN model including the martingale constraint (AJ-Φ_ξ), and the AJ-GAN model in which the transitory component is constructed using only bond data ($\text{AJ}^{(\text{bond})}$). The cumulative return is reset to zero on the transition of each subsample to the next, i.e. from training data to validation data and from validation data to testing data. Dotted lines corresponds to the transitory component and solid lines corresponds to the permanent component. All time series are scaled to have an annualized volatility of 20%.

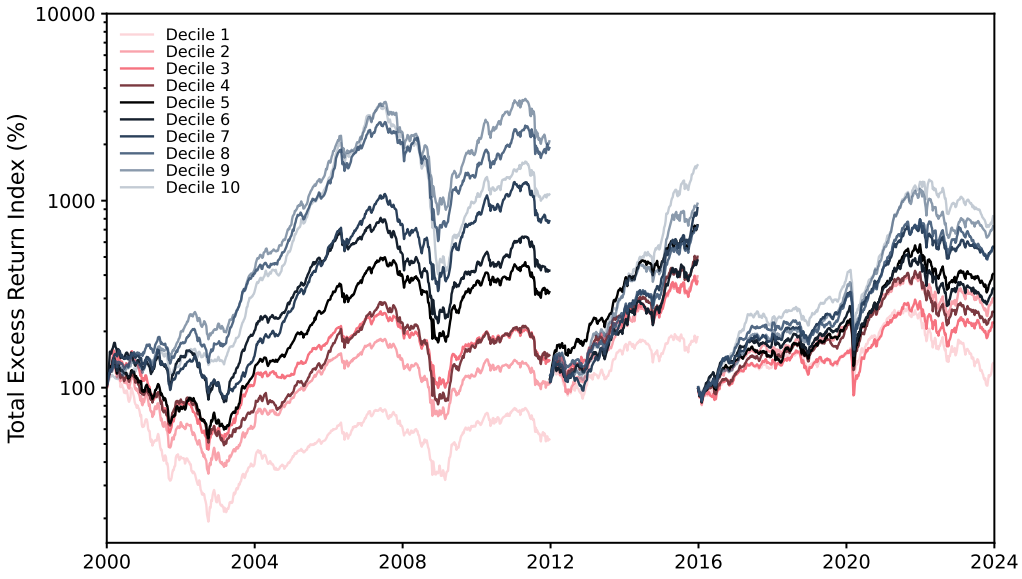


Figure A2

Predictive Performance of the GAN Model.

Notes. The figure illustrates the cumulative excess return index $100 \cdot \left(\prod_{t=1}^{T_{\text{sample}}} \left(1 + R^e_{t+1,d}\right)\right)$ of the 10 decile-sorted and equally weighted portfolios (EWP) based on loadings $\beta_{t,i}$. The loadings are those generated by the generative adversarial network (GAN) and d denotes a specific decile. The EWPs are rebalanced every week, and the index is reset on the transition from one sub-sample to the other. The training set is from January 2000 to December 2011, the validation set is from December 2011 to December 2016, and the test set is from December 2016 to December 2023. The loadings are scaled every week, so the loading of the pricing kernel is one; $\hat{\omega}_t' \hat{\beta}_t = 1$.

Table A1

Correlation of Pricing Kernel Components to the Market Portfolio

	$\rho\left(M^{(\text{T})}_{t+1}, M^{(\text{CAPM})}_{t+1}\right)$			$\rho\left(M^{(\text{P})}_{t+1}, M^{(\text{CAPM})}_{t+1}\right)$			$\rho\left(M_{t+1}, M^{(\text{CAPM})}_{t+1}\right)$		
Model	Train	Valid	Test	Train	Valid	Test	Train	Valid	Test
AJ-GAN	0.55	0.50	0.67	-0.33	0.10	0.08	0.18	0.40	0.54
AJ-Φ_ξ	0.43	0.66	0.64	-0.60	-0.76	-0.59	-0.32	-0.21	0.07
AJ$^{(\text{bond})}$	-0.11	0.06	0.15	-0.18	0.04	0.36	-0.20	0.05	0.39

Notes. This table illustrates the weekly correlation (ρ) of each pricing kernel M_{t+1} component with the pricing kernel implied by market weighted portfolio of all stocks $M^{(\text{CAPM})}_{t+1}$. The models presented are the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ model with martingale regularization (AJ-Φ_ξ), and the AJ-GAN model using bond index data for the transitory component (AJ$^{(\text{bond})}$). The components include the transitory component $M^{(\text{T})}_{t+1}$, the permanent component $M^{(\text{P})}_{t+1}$, and the full pricing kernel M_{t+1}. The training data covers 2000-2011, the validation data covers 2012-2015, and the test data covers 2016-2023

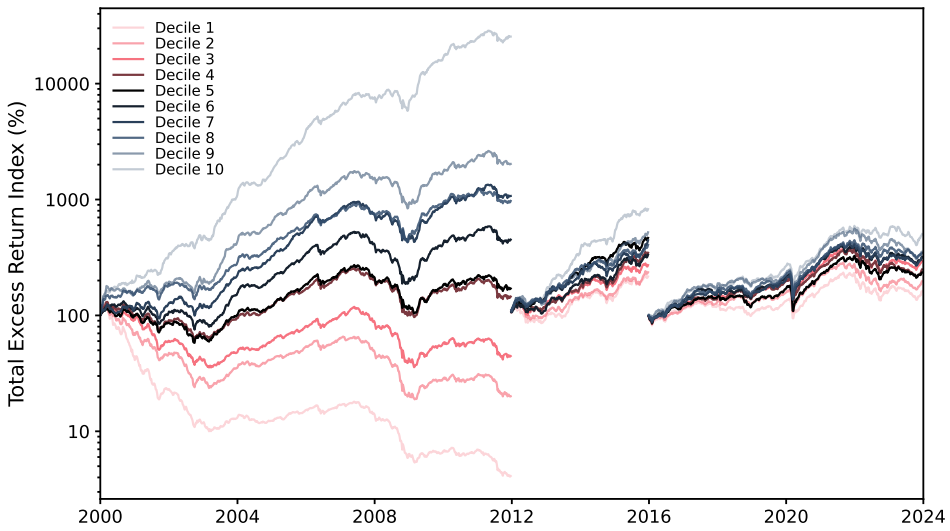


Figure A3

Predictive Performance of the FFN Model.

Notes. The figure illustrates the cumulative excess return index $100 \cdot \left(\prod_{t=1}^{T_{\text{sample}}} \left(1 + R^e_{t+1,d}\right)\right)$ of the 10 decile-sorted and equally weighted portfolios (EWP) based on loadings $\beta_{t,i}$. The loadings are those generated by the feedforward network (FFN) and d denotes a specific decile. The EWPs are rebalanced every week, and the index is reset on the transition from one subsample to the other. The training set is from January 2000 to December 2011, the validation set is from December 2011 to December 2016, and the test set is from December 2016 to December 2023. The loadings are scaled every week, so the loading of the pricing kernel is one; $\hat{\omega}_t' \hat{\beta}_t = 1$.

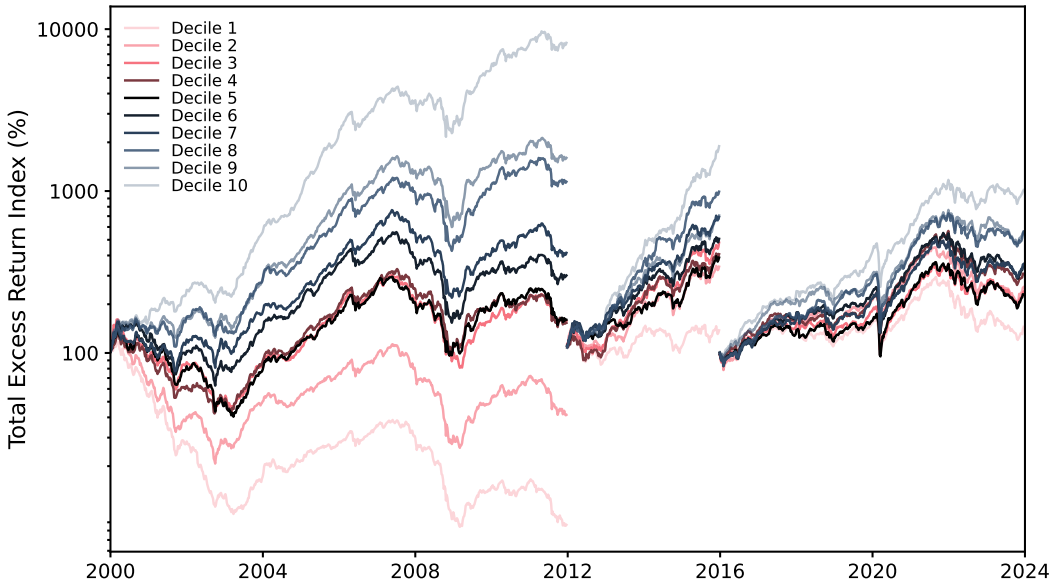


Figure A4

Predictive Performance of the EN Model.

Notes. The figure illustrates the cumulative excess return index $100 \cdot \left(\prod_{t=1}^{T_{\text{sample}}} \left(1 + R^e_{t+1,d}\right)\right)$ of the 10 decile-sorted and equally weighted portfolios (EWP) based on loadings $\beta_{t,i}$. The loadings are those generated by the elastic net (EN) and d denotes a specific decile. The EWPs are rebalanced every week, and the index is reset on the transition from one subsample to the other. The training set is from January 2000 to December 2011, the validation set is from December 2011 to December 2016, and the test set is from December 2016 to December 2023. The loadings are scaled every week, so the loading of the pricing kernel is one; $\hat{\omega}_t' \hat{\beta}_t = 1$.

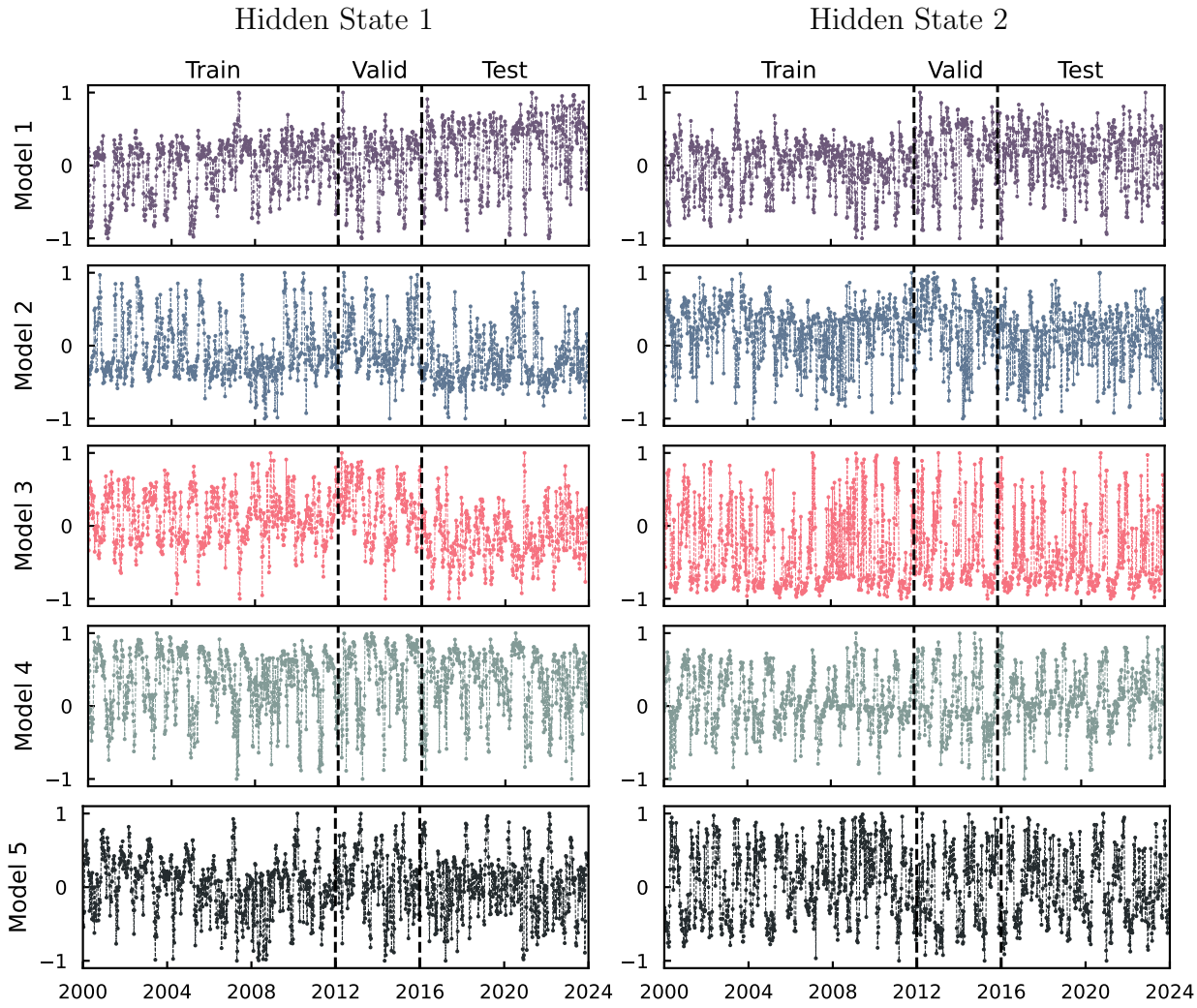


Figure A5

LSTM-Estimated Macroeconomic Hidden State Process h_t by GAN Model

Notes. This figure illustrates the estimated weekly hidden states h_t. Five estimates are produced from each long- and short-term memory (LSTM) model within the 5 pricing kernel (PK) networks in the ensemble generative adversarial network (GAN) model. Each color corresponds to one component model of the final model ensemble. All hidden states are normalized to $[-1, 1]$ during the training process and during model evaluation. Is not necessarily the same economic dynamics represented across model fits. Therefore, hidden states are not aggregated for the model ensemble.

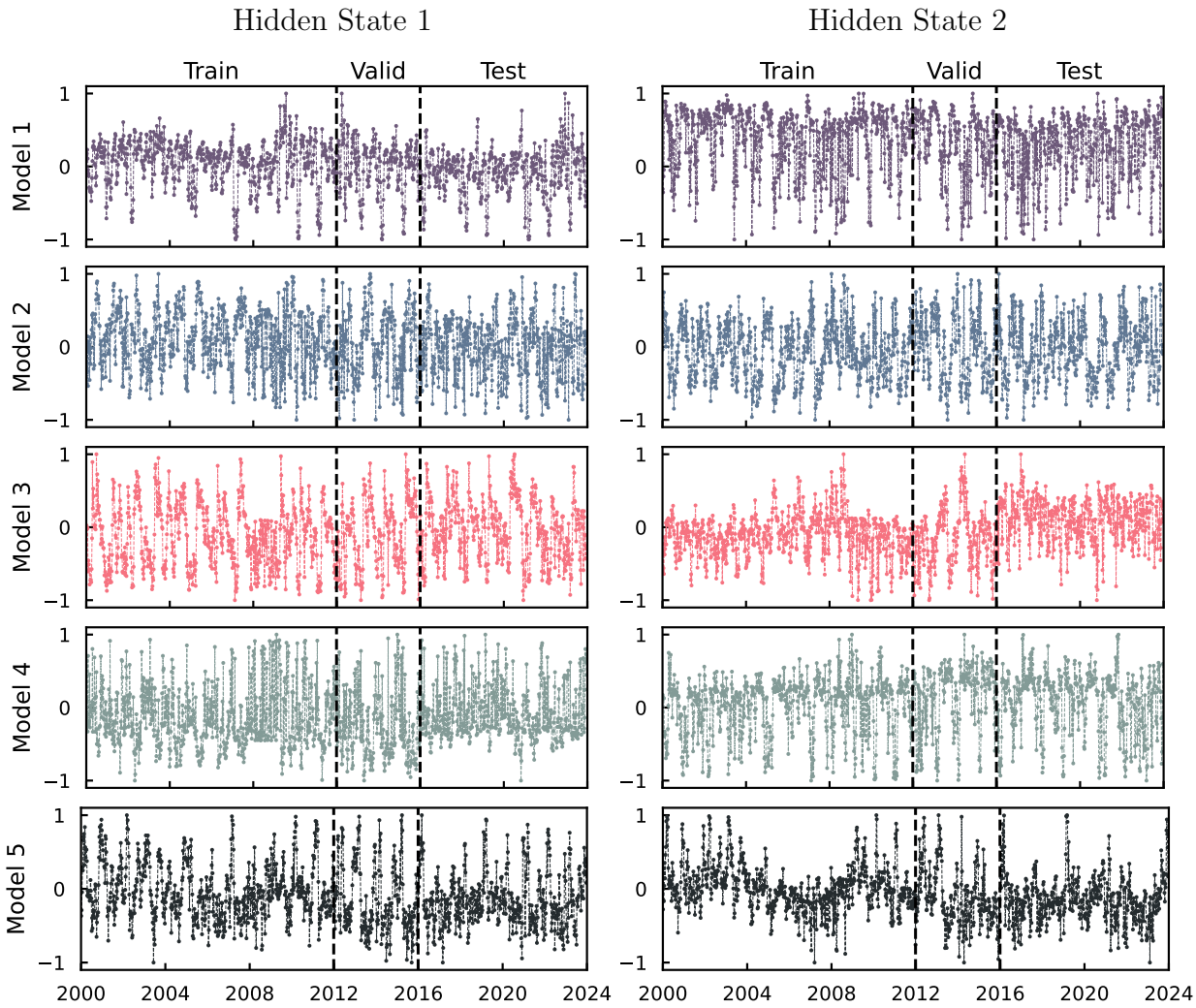


Figure A6

LSTM-Estimated Macroeconomic Hidden State Process h_t by the AJ-GAN Model

Notes. This figure illustrates the estimated weekly hidden states h_t from the LSTM model within the $\xi = 0$ AJ-GAN Model. Note that it is not necessarily the same economic dynamics that is captured in each fitted state across model fits, which is why hidden states are not themselves aggregated across models. The dashed lines separate the training, validation, and test datasets.

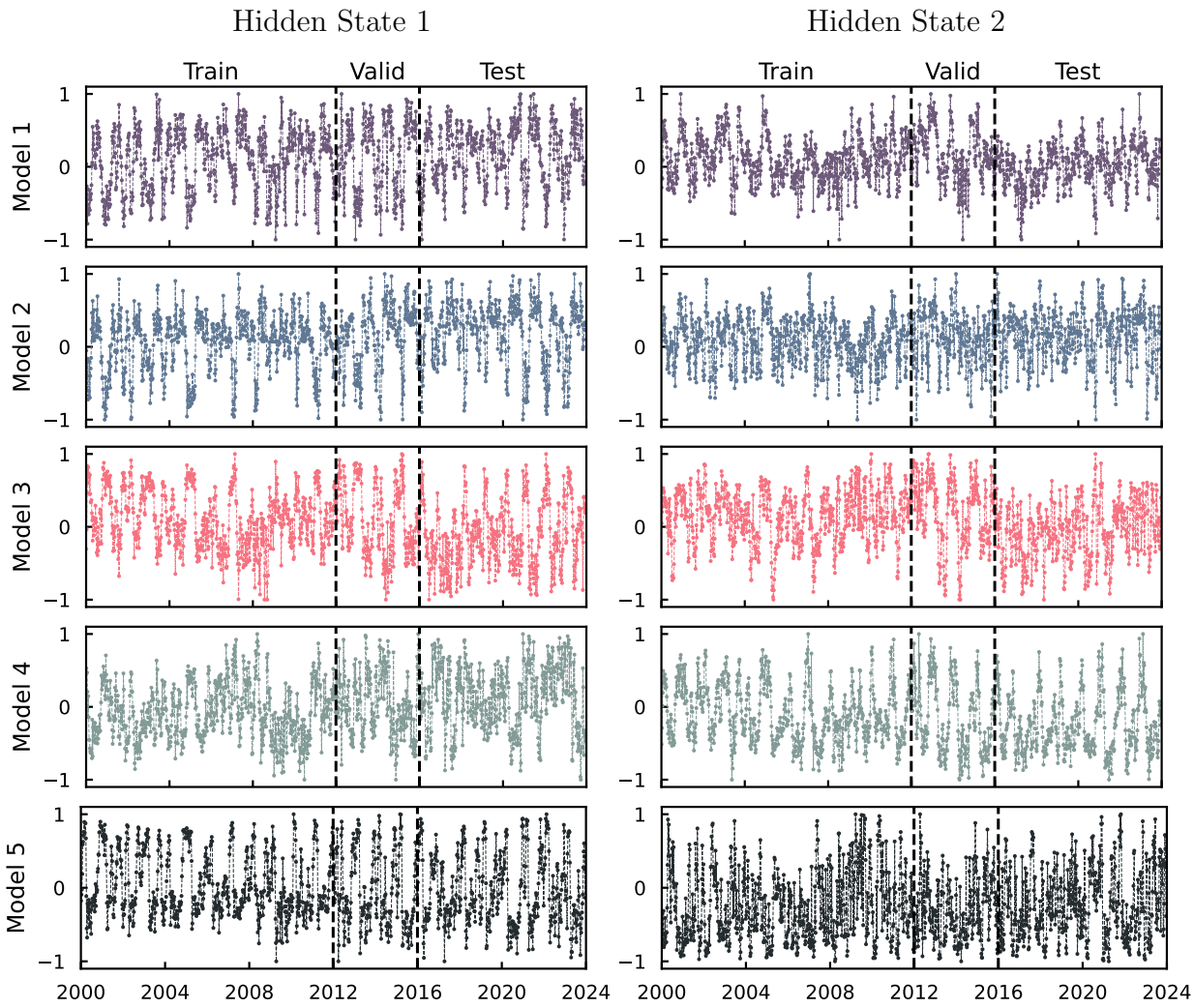


Figure A7

LSTM-Estimated Macroeconomic Hidden State Process h_t by the AJ-Φ_ξ Model

Notes. This figure illustrates the estimated weekly hidden states h_t from the LSTM model within the AJ-Φ_ξ Model. Note that it is not necessarily the same economic dynamics that is captured in each fitted state across model fits, which is why hidden states are not themselves aggregated across models. The dashed lines separate the training, validation, and test datasets.

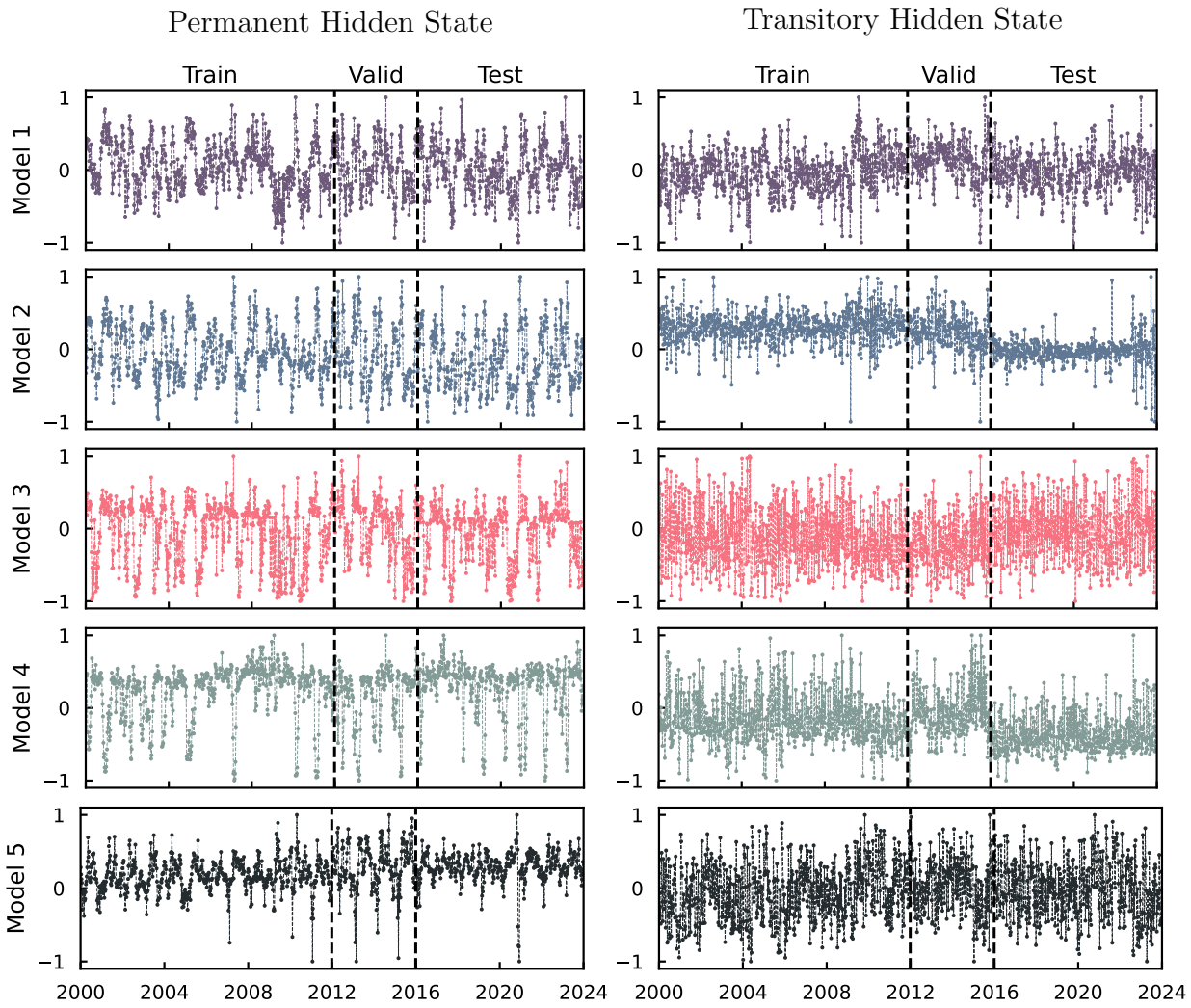


Figure A8

LSTM-Estimated Macroeconomic Hidden State Process h_t by the AJ$^{(\text{bond})}$ Model

Notes. This figure illustrates the estimated weekly hidden states h_t from the LSTM model within the AJ$^{(\text{bond})}$ Model. Figures on the left are the estimated hidden states from the permanent pricing kernel (PK) network, and figures on the right are from the transitory PK network. In the AJ$^{(\text{bond})}$ model, the permanent PK network uses only stock data, and the transitory network uses only data on aggregate bond indices. Note that it is not necessarily the same economic dynamics that is captured in each fitted state across model fits, which is why hidden states are not themselves aggregated across models. The dashed lines separate the training, validation, and test datasets.

A2. Additional Characteristic Pricing Tables

Table A2
Pricing the Most Important Characteristic Portfolios (according to FFN)

	Total Assets (AT)						Assets to Market Cap. (A2ME)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN
1	0.69	0.68	-0.02	0.10	-0.03	0.41	0.48	0.08	0.15	-0.00	0.19	0.36
2	0.77	0.58	0.26	0.21	0.18	0.62	0.63	-0.16	0.35	-0.14	0.16	0.18
3	0.81	0.23	0.13	-0.09	0.02	0.21	0.77	0.19	0.25	0.03	0.14	0.10
4	0.82	0.42	0.11	0.01	0.15	0.37	0.73	0.23	0.34	-0.02	0.20	0.06
5	0.84	0.26	0.14	-0.02	0.15	0.30	0.73	0.42	0.34	0.04	0.23	0.29
6	0.78	0.15	0.25	0.04	0.11	0.45	0.78	0.61	0.45	-0.08	-0.02	0.06
7	0.86	0.28	0.33	-0.14	-0.02	0.20	0.81	0.69	0.43	0.03	0.15	0.11
8	0.81	0.13	0.34	-0.10	0.08	0.20	0.77	0.69	0.44	0.24	0.18	0.44
9	0.84	0.38	0.36	-0.01	0.14	0.21	0.73	0.58	0.24	0.05	-0.08	0.08
10	0.64	0.39	0.34	0.01	0.08	0.26	0.58	0.45	0.42	0.03	0.00	0.35
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.79	0.37	0.21	0.91	0.87	-0.22	0.69	0.40	0.34	0.91	0.77	0.40
	Market Capitalization (LME)						Tobin's Q (Q)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN
1	0.45	0.18	-0.01	0.03	-0.16	0.34	0.54	0.39	0.24	0.09	0.30	0.28
2	0.74	0.76	-0.06	-0.07	-0.09	0.21	0.64	0.33	0.16	-0.13	-0.10	0.12
3	0.81	0.80	-0.08	-0.11	-0.07	0.21	0.70	0.24	0.39	0.03	0.18	0.09
4	0.88	0.74	-0.01	-0.03	0.16	0.32	0.64	0.23	0.38	0.08	0.23	0.34
5	0.92	0.62	0.01	-0.02	0.15	0.35	0.72	0.37	0.37	-0.02	0.20	0.12
6	0.91	0.43	0.01	0.03	0.29	0.33	0.72	0.26	0.38	0.04	0.22	0.27
7	0.85	0.31	0.09	0.05	0.25	0.36	0.85	0.32	0.42	-0.02	0.20	0.21
8	0.81	0.31	0.11	0.08	0.26	0.34	0.81	0.42	0.29	0.03	0.20	0.16
9	0.79	0.35	0.06	0.05	0.22	0.32	0.85	0.45	0.28	-0.06	0.16	0.06
10	0.74	0.27	0.24	0.06	0.22	0.31	0.58	0.39	0.41	-0.07	0.06	0.21
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.81	0.46	0.04	0.97	0.60	0.02	0.69	0.34	0.32	0.95	0.62	0.58

Notes. This table records out-of-sample (OOS) metrics of time series explained variation (TS-R^2) and weekly pricing errors (α) on 40 characteristically managed portfolios. TS-R^2 and explained variation in average returns (CS-R^2) are presented across all deciles on row "All". Pricing errors are expressed in weekly percentage units. The characteristically managed portfolios are constructed by sorting stocks into deciles for each characteristic. Portfolios are value-weighted and rebalanced weekly within decile. The characteristics presented here are the 4 most important characteristics as determined by the saliency analysis for the FFN model in Figure VI. These characteristics are total assets (AT), assets to market capitalization (A2ME), market capitalization (LME), and Tobin's Q (Q).

Table A3

Pricing the Most Important Characteristic Portfolios (according to EN)

	Standard Unexplained Variation (SUV)						Turnover (Lturnover)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN
1	0.70	-0.23	0.57	0.11	0.48	0.22	0.43	-0.68	0.05	-0.34	0.17	-0.03
2	0.71	-0.18	0.48	-0.03	0.25	0.20	0.70	-0.00	0.16	0.04	0.35	0.26
3	0.74	0.06	0.35	0.01	0.19	0.18	0.72	0.03	0.29	-0.04	0.34	0.01
4	0.79	0.14	0.27	0.12	0.42	0.34	0.64	0.06	0.36	0.13	0.32	0.25
5	0.74	0.18	0.17	0.00	0.24	0.25	0.63	0.14	0.34	-0.09	0.16	0.09
6	0.77	0.33	0.01	0.00	0.26	0.30	0.68	0.15	0.36	0.17	0.27	0.30
7	0.72	0.27	-0.10	-0.04	0.09	0.30	0.76	0.33	0.24	-0.03	0.16	0.21
8	0.75	0.45	-0.22	-0.06	0.04	0.25	0.73	0.45	0.16	0.12	0.17	0.24
9	0.72	0.53	-0.36	-0.05	0.09	0.32	0.70	0.42	0.02	0.12	0.16	0.23
10	0.71	0.61	-0.46	-0.01	0.06	0.40	0.71	0.41	0.01	0.21	0.22	0.30
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.73	0.21	0.09	0.97	0.35	0.19	0.68	0.16	0.19	0.76	0.41	0.53
	FCF to Equity (CF)						Market Capitalization (LME)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN	GAN	FFN	EN
1	0.50	0.45	0.31	-0.04	0.22	0.16	0.45	0.18	-0.01	0.03	-0.16	0.34
2	0.65	0.15	0.15	-0.18	0.03	-0.19	0.74	0.76	-0.06	-0.07	-0.09	0.21
3	0.67	0.23	0.32	0.06	0.24	0.09	0.81	0.80	-0.08	-0.11	-0.07	0.21
4	0.75	0.42	0.33	0.05	0.18	0.28	0.88	0.74	-0.01	-0.03	0.16	0.32
5	0.78	0.22	0.25	-0.00	0.03	0.13	0.92	0.62	0.01	-0.02	0.15	0.35
6	0.81	0.32	0.42	-0.09	0.05	0.13	0.91	0.43	0.01	0.03	0.29	0.33
7	0.78	0.30	0.30	0.12	0.24	0.32	0.85	0.31	0.09	0.05	0.25	0.36
8	0.80	0.34	0.19	-0.07	0.10	0.16	0.81	0.31	0.11	0.08	0.26	0.34
9	0.72	0.39	0.22	0.13	0.24	0.26	0.79	0.35	0.06	0.05	0.22	0.32
10	0.50	0.25	0.06	0.04	0.21	0.34	0.74	0.27	0.24	0.06	0.22	0.31
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.68	0.31	0.26	0.91	0.69	0.51	0.81	0.46	0.04	0.97	0.60	0.02

Notes. This table records out-of-sample (OOS) metrics of time series explained variation (TS-R^2) and weekly pricing errors (α) on 40 characteristically managed portfolios. TS-R^2 and explained variation in average returns (CS-R^2) are presented across all deciles on row "All". Pricing errors are expressed in weekly percentage units. The characteristically managed portfolios are constructed by sorting stocks into deciles for each characteristic. Portfolios are value-weighted and rebalanced weekly within decile. The characteristics presented here are the 4 most important characteristics as determined by the saliency analysis for the EN model in Figure VI. These characteristics are standard unexplained variation (SUV), turnover (Lturnover), FCF to equity (CF), and market capitalization (LME).

Table A4

Pricing the Most Important Characteristic Portfolios (according to AJ-GAN)

	Market Capitalization (LME)						Standard Unexplained Variation (SUV)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$
1	0.57	0.48	0.56	0.11	0.06	0.12	0.73	0.73	0.74	0.12	0.11	0.10
2	0.76	0.75	0.77	-0.01	-0.05	-0.00	0.73	0.73	0.72	0.02	0.01	0.03
3	0.83	0.82	0.83	-0.08	-0.09	-0.07	0.76	0.75	0.76	-0.01	-0.02	-0.00
4	0.88	0.88	0.88	0.01	0.00	0.02	0.81	0.80	0.81	0.13	0.14	0.13
5	0.92	0.92	0.92	-0.01	-0.00	0.01	0.76	0.76	0.77	-0.00	0.00	0.00
6	0.92	0.91	0.92	0.03	0.04	0.04	0.78	0.77	0.77	-0.01	0.00	0.01
7	0.90	0.88	0.91	0.02	0.03	0.02	0.74	0.74	0.75	-0.05	-0.05	-0.05
8	0.90	0.87	0.91	0.00	0.03	-0.01	0.76	0.76	0.76	-0.11	-0.10	-0.12
9	0.90	0.87	0.90	-0.04	-0.01	-0.05	0.75	0.75	0.76	-0.06	-0.06	-0.06
10	0.75	0.77	0.76	-0.05	-0.01	-0.04	0.74	0.74	0.75	-0.04	-0.03	-0.02
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.85	0.83	0.85	0.98	0.98	0.97	0.76	0.75	0.76	0.95	0.95	0.95
	Net Share Issues (NSI)						Price-to-Cost (PCM)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$
1	0.70	0.70	0.72	0.10	0.10	0.14	0.64	0.65	0.68	0.01	0.00	-0.03
2	0.81	0.81	0.81	-0.01	-0.01	0.01	0.74	0.73	0.74	0.01	0.01	0.00
3	0.76	0.76	0.75	0.01	0.00	0.02	0.73	0.73	0.74	-0.06	-0.05	-0.07
4	0.82	0.83	0.83	-0.14	-0.14	-0.12	0.77	0.76	0.78	0.05	0.06	0.05
5	0.71	0.71	0.71	0.01	0.01	-0.00	0.79	0.78	0.80	-0.00	-0.01	-0.01
6	0.75	0.74	0.74	0.01	0.02	0.00	0.82	0.82	0.82	-0.04	-0.03	-0.04
7	0.76	0.76	0.76	-0.05	-0.06	-0.05	0.81	0.82	0.81	-0.12	-0.12	-0.10
8	0.76	0.75	0.77	-0.16	-0.16	-0.14	0.73	0.72	0.73	-0.06	-0.05	-0.04
9	0.57	0.56	0.58	0.30	0.29	0.28	0.45	0.44	0.45	-0.03	-0.04	-0.01
10	0.73	0.71	0.76	-0.08	-0.07	-0.09	0.57	0.57	0.56	0.21	0.22	0.28
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.73	0.73	0.74	0.85	0.85	0.86	0.69	0.69	0.70	0.93	0.92	0.90

Notes. This table records out-of-sample (OOS) metrics of time series R^2 and weekly pricing errors α (in percentage units) for all decile-sorted portfolios. The metrics are presented for the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN including the martingale constraint (AJ-Φ_ξ), and the AJ-GAN model in which the transitory component is built only using bond data AJ$^{(\text{bond})}$. The characteristics used are the four most important characteristics determined by the saliency analysis for the combined AJ-GAN pricing kernel presented in Figure VII. These characteristics are market capitalization (LME), standard unexplained variation (SUV), net share issue (NSI), and price-to-cost margin (PCM).

Table A5

Pricing the Most Important Characteristic Portfolios (according to AJ-Φ_ξ)

	Long-Term Momentum (r36_3)						Turnover (Lturnover)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$
1	0.58	0.61	0.60	0.29	0.25	0.27	0.61	0.58	0.61	-0.24	-0.26	-0.24
2	0.67	0.68	0.68	0.07	0.03	0.04	0.72	0.72	0.73	0.11	0.09	0.10
3	0.71	0.71	0.70	-0.02	-0.04	-0.01	0.75	0.74	0.75	-0.04	-0.04	-0.03
4	0.79	0.79	0.79	-0.06	-0.07	-0.06	0.67	0.65	0.68	0.11	0.11	0.11
5	0.76	0.76	0.77	0.14	0.13	0.12	0.65	0.64	0.65	-0.12	-0.12	-0.10
6	0.72	0.73	0.73	-0.10	-0.11	-0.09	0.70	0.69	0.70	0.11	0.12	0.10
7	0.77	0.76	0.77	-0.05	-0.05	-0.02	0.79	0.79	0.79	-0.07	-0.05	-0.03
8	0.72	0.70	0.74	0.03	0.05	0.04	0.79	0.78	0.80	0.03	0.04	0.03
9	0.64	0.60	0.65	-0.19	-0.19	-0.19	0.81	0.80	0.83	-0.01	-0.00	-0.01
10	0.64	0.59	0.67	-0.05	-0.03	-0.05	0.84	0.83	0.86	0.09	0.11	0.09
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.68	0.67	0.70	0.84	0.86	0.85	0.74	0.73	0.75	0.87	0.87	0.89
	Operating Accrual (OA)						Net Operating Assets (NOA)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$	AJ	AJ-Φ	AJ$^{(b)}$
1	0.62	0.62	0.62	0.04	0.03	0.02	0.59	0.57	0.58	-0.15	-0.14	-0.18
2	0.62	0.62	0.63	-0.22	-0.22	-0.24	0.73	0.74	0.74	0.19	0.18	0.18
3	0.71	0.71	0.71	0.03	0.02	0.01	0.76	0.75	0.76	0.03	0.03	0.04
4	0.63	0.63	0.64	0.12	0.12	0.13	0.81	0.81	0.81	-0.14	-0.14	-0.13
5	0.71	0.70	0.71	-0.06	-0.05	-0.05	0.73	0.73	0.74	0.21	0.19	0.21
6	0.77	0.76	0.76	-0.01	-0.00	0.03	0.81	0.81	0.81	0.03	0.03	0.04
7	0.77	0.77	0.78	0.04	0.05	0.06	0.82	0.82	0.82	-0.01	-0.01	-0.00
8	0.76	0.76	0.77	0.03	0.04	0.05	0.83	0.83	0.84	-0.07	-0.06	-0.05
9	0.81	0.80	0.81	-0.03	-0.01	-0.02	0.87	0.87	0.87	-0.09	-0.08	-0.08
10	0.80	0.79	0.79	0.03	0.04	0.04	0.66	0.68	0.68	-0.01	0.00	-0.01
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.72	0.72	0.72	0.93	0.93	0.91	0.76	0.76	0.76	0.86	0.88	0.86

Notes. This table records out-of-sample (OOS) metrics of time series R^2 and weekly pricing errors α (in percentage units) for all decile-sorted portfolios. The metrics are presented for the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN including the martingale constraint (AJ-Φ_ξ), and the AJ-GAN model in which the transitory component is built only using bond data AJ$^{(\text{bond})}$. The characteristics used are the four most important characteristics determined by the saliency analysis for the combined AJ-Φ_ξ pricing kernel presented in Figure VII. These characteristics are long-term momentum (r36_3), turnover (Lturnover), operating accrual (OA), and net operating assets (NOA).

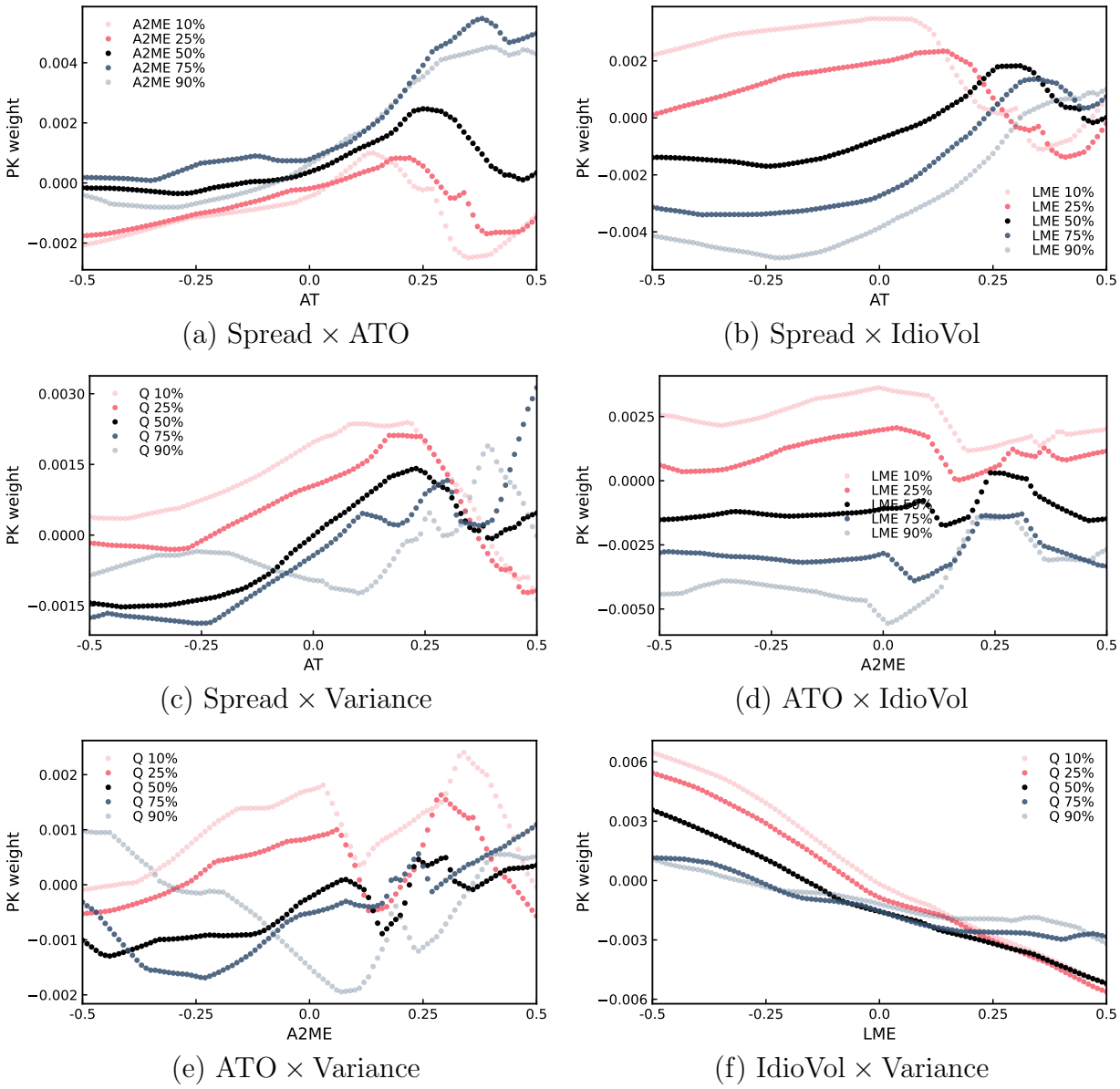
Table A6

Pricing the Most Important Characteristic Portfolios (according to $\text{AJ}^{(\text{bond})}$)

	Turnover (Lturnover)						Standard Unexplained Variation (SUV)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$
1	0.61	0.58	0.61	-0.24	-0.26	-0.24	0.73	0.73	0.74	0.12	0.11	0.10
2	0.72	0.72	0.73	0.11	0.09	0.10	0.73	0.73	0.72	0.02	0.01	0.03
3	0.75	0.74	0.75	-0.04	-0.04	-0.03	0.76	0.75	0.76	-0.01	-0.02	-0.00
4	0.67	0.65	0.68	0.11	0.11	0.11	0.81	0.80	0.81	0.13	0.14	0.13
5	0.65	0.64	0.65	-0.12	-0.12	-0.10	0.76	0.76	0.77	-0.00	0.00	0.00
6	0.70	0.69	0.70	0.11	0.12	0.10	0.78	0.77	0.77	-0.01	0.00	0.01
7	0.79	0.79	0.79	-0.07	-0.05	-0.03	0.74	0.74	0.75	-0.05	-0.05	-0.05
8	0.79	0.78	0.80	0.03	0.04	0.03	0.76	0.76	0.76	-0.11	-0.10	-0.12
9	0.81	0.80	0.83	-0.01	-0.00	-0.01	0.75	0.75	0.76	-0.06	-0.06	-0.06
10	0.84	0.83	0.86	0.09	0.11	0.09	0.74	0.74	0.75	-0.04	-0.03	-0.02
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.74	0.73	0.75	0.87	0.87	0.89	0.76	0.75	0.76	0.95	0.95	0.95
	Residual Variance (Resid_Var)						Market Capitalization (LME)					
	Time-Series R^2			Pricing Error (α)			Time-Series R^2			Pricing Error (α)		
Decile	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$	AJ	AJ-Φ	$\text{AJ}^{(\text{b})}$
1	0.50	0.49	0.60	0.16	0.17	0.21	0.57	0.48	0.56	0.11	0.06	0.12
2	0.66	0.67	0.71	0.03	0.04	0.05	0.76	0.75	0.77	-0.01	-0.05	-0.00
3	0.75	0.75	0.76	-0.07	-0.06	-0.06	0.83	0.82	0.83	-0.08	-0.09	-0.07
4	0.80	0.80	0.79	0.04	0.04	0.06	0.88	0.88	0.88	0.01	0.00	0.02
5	0.78	0.77	0.77	0.03	0.04	0.04	0.92	0.92	0.92	-0.01	-0.00	0.01
6	0.79	0.79	0.80	0.04	0.05	0.02	0.92	0.91	0.92	0.03	0.04	0.04
7	0.76	0.75	0.77	0.02	0.02	0.01	0.90	0.88	0.91	0.02	0.03	0.02
8	0.73	0.73	0.75	-0.08	-0.09	-0.10	0.90	0.87	0.91	0.00	0.03	-0.01
9	0.67	0.67	0.68	0.07	0.06	0.05	0.90	0.87	0.90	-0.04	-0.01	-0.05
10	0.57	0.57	0.60	-0.25	-0.29	-0.27	0.75	0.77	0.76	-0.05	-0.01	-0.04
	Time-Series R^2			Cross-Sectional R^2			Time-Series R^2			Cross-Sectional R^2		
All	0.70	0.70	0.72	0.89	0.86	0.86	0.85	0.83	0.85	0.98	0.98	0.97

Notes. This table records out-of-sample (OOS) metrics of time series R^2 and weekly pricing errors α (in percentage units) for all decile-sorted portfolios. The metrics are presented for the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN including the martingale constraint (AJ-Φ_ξ), and the AJ-GAN model in which the transitory component is built only using bond data $\text{AJ}^{(\text{bond})}$. The characteristics used are the four most important characteristics determined by the saliency analysis for the combined $\text{AJ}^{(\text{bond})}$ pricing kernel presented in Figure VIII. These characteristics are turnover (Lturnover), standard unexplained variation (SUV), residual variance (Resid_Var), and market capitalization (LME).

A3. Additional Characteristic Interactions



(a) Spread × ATO

(b) Spread × IdioVol

(c) Spread × Variance

(d) ATO × IdioVol

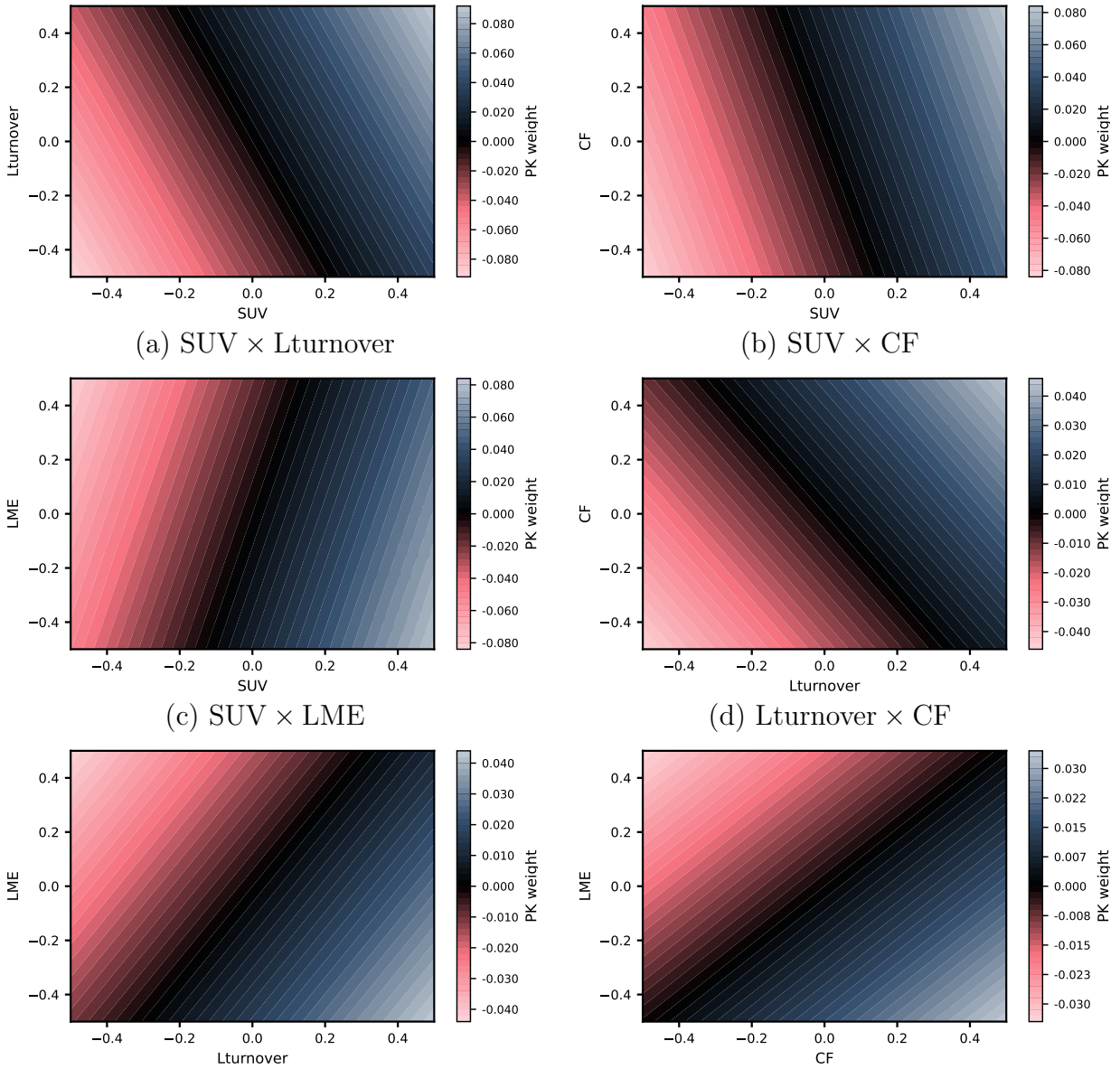
(e) ATO × Variance

(f) IdioVol × Variance

Figure A9

Characteristic Interactions within FFN Pricing Kernel $\hat{\omega}_{FFN}$

Notes. The figure illustrates the out-of-sample interactions of predictors in the ensemble feedforward network (FFN) pricing kernel $\hat{\omega}_{\text{FFN}}$ (before $\|\omega_t\|_1$ normalization). The characteristics shown are the four most important characteristics according to the saliency analysis in Figure VI, i.e., total assets (AT), assets to market capitalization (A2ME), market capitalization (LME), and Tobin's Q (Q). The interacted variables are set to the 10th, 25th, 50th, 75th, and 90th quantiles. Other firm characteristics are kept constant at median levels.



(a) SUV × Lturnover

(b) SUV × CF

(c) SUV × LME

(d) Lturnover × CF

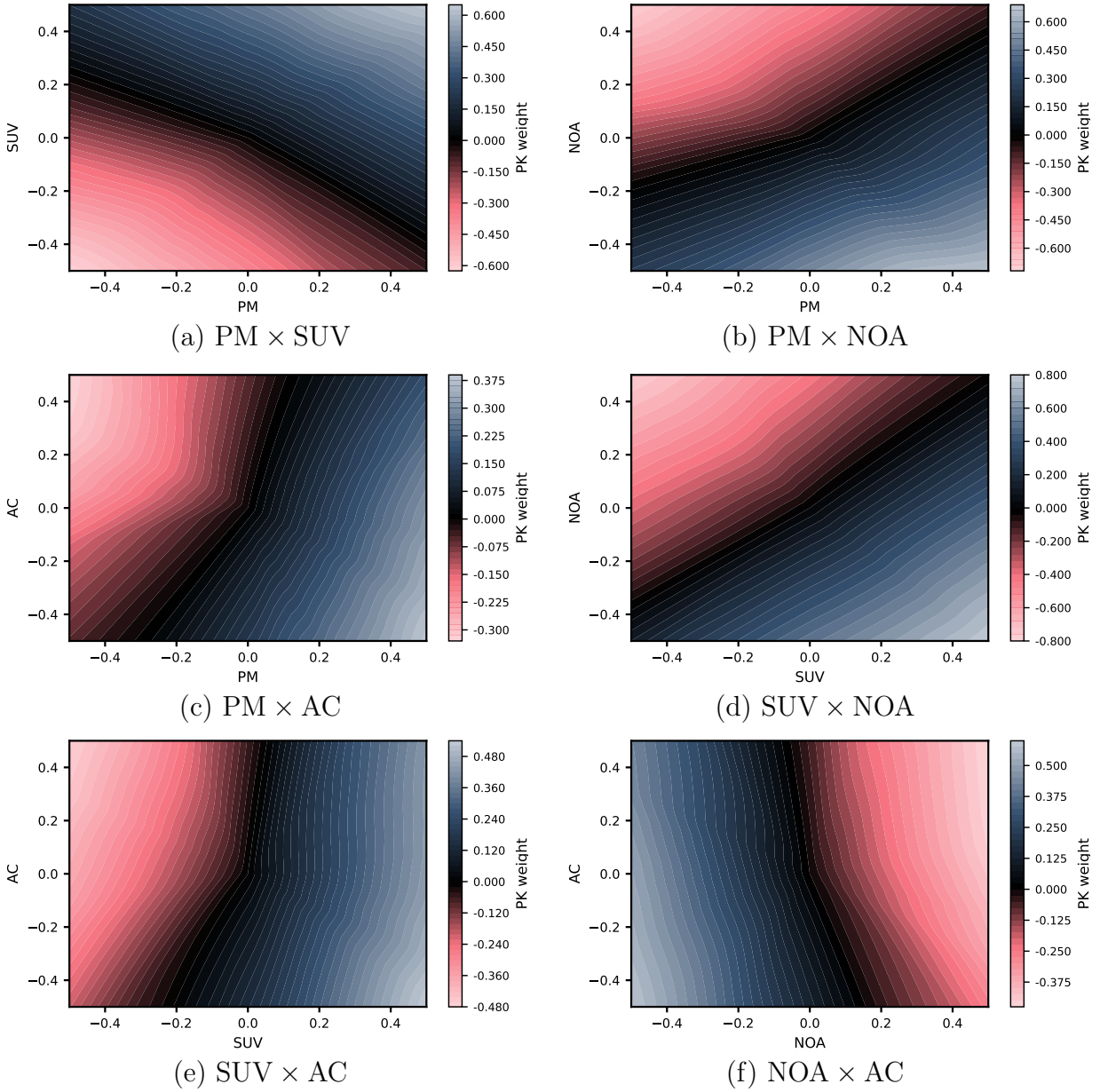
(e) Lturnover × LME

(f) CF × LME

Figure A10

Elastic Net Weights ω_{EN} as a Function of Two Characteristics

Notes. The figure illustrates the out-of-sample non-linear effect on the elastic net (EN) pricing kernel portfolio $\hat{\omega}_{\text{EN}}$ (before $\|\omega_t\|_1$ normalization) for interactions between the four most important firm characteristics from Figure VI, i.e. standard unexplained variation (SUV), turnover (Lturnover), free cash flow to equity (CF), and market capitalization (LME). Other firm characteristics are held constant at median levels.



(a) PM × SUV (b) PM × NOA

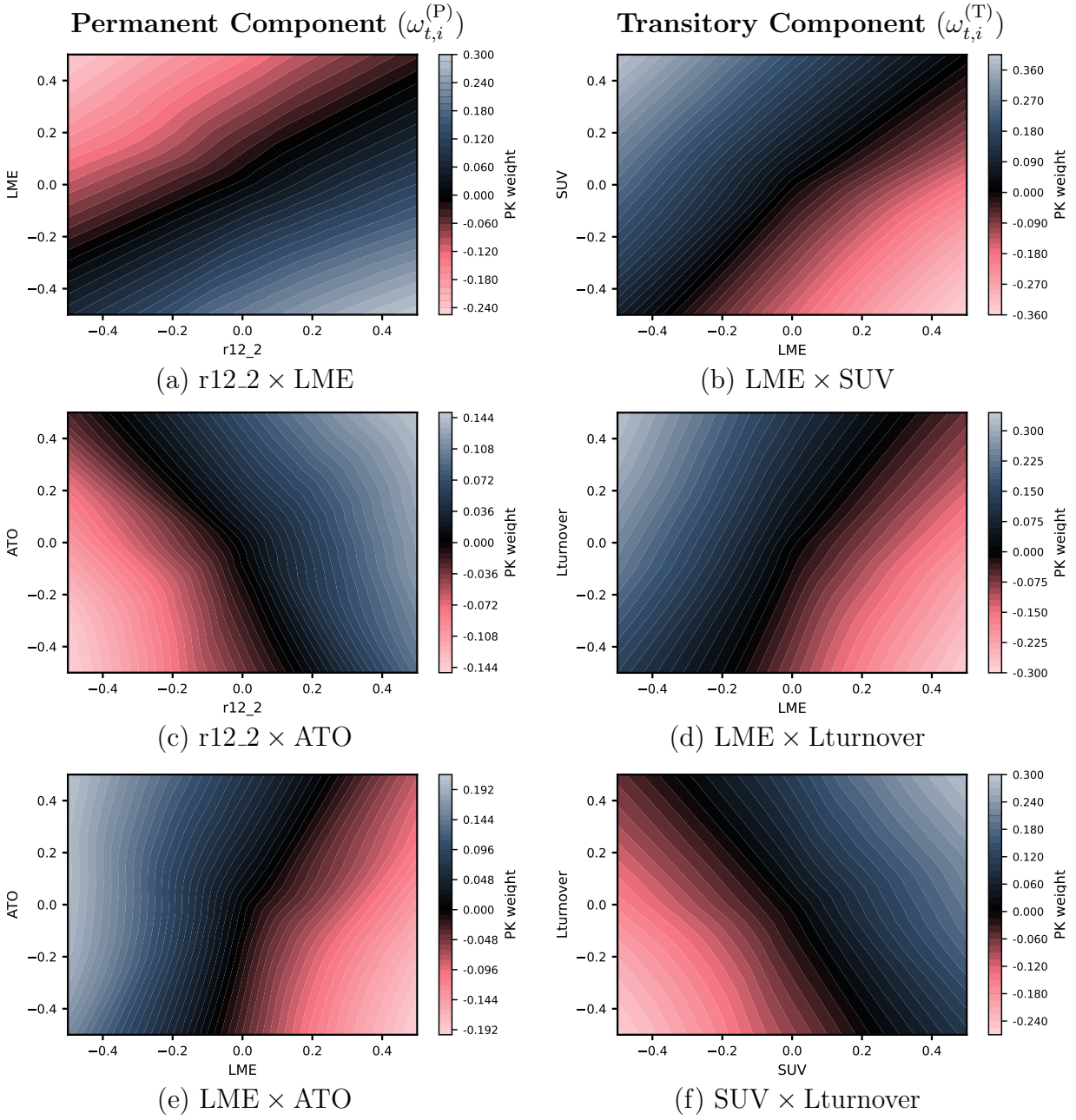
(c) PM × AC (d) SUV × NOA

(e) SUV × AC (f) NOA × AC

Figure A11

GAN Weights ω_{GAN} as a Function of Two Characteristics

Notes. The figure illustrates the out-of-sample non-linear effect on the generative adversarial network (GAN) pricing kernel $\hat{\omega}_{\text{GAN}}$ (before $\|\omega_t\|_1$ normalization) for interactions between the four most important firm characteristics from Figure VI, i.e. profit margin (PM), standard unexplained variation (SUV), net operating assets (NOA), and accruals (AC). All other firm characteristics are kept constant at median levels, and both hidden states are fixed at zero.



(a) r12_2 × LME (b) LME × SUV

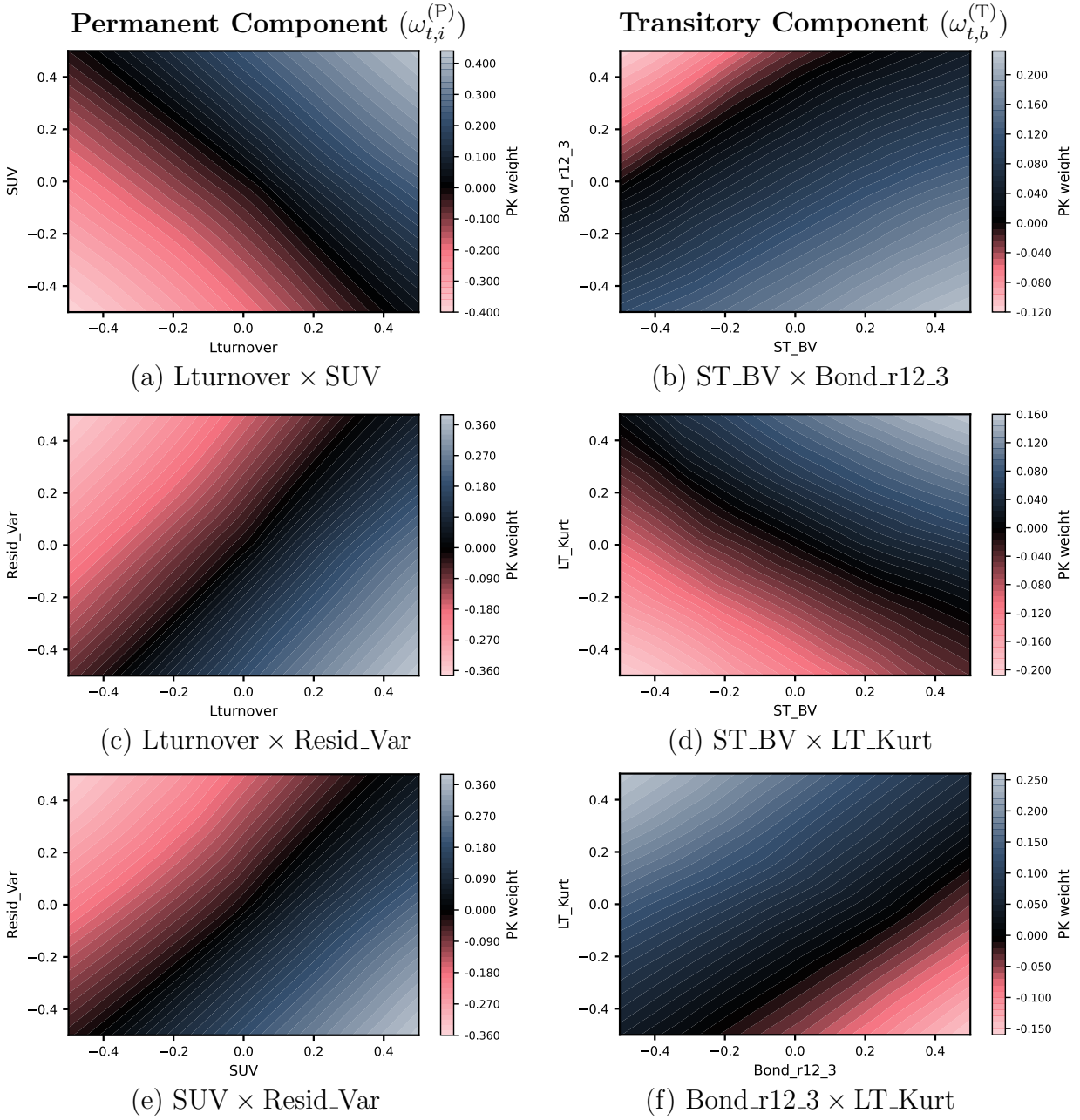
(c) r12_2 × ATO (d) LME × Lturnover

(e) LME × ATO (f) SUV × Lturnover

Figure A12

AJ-GAN Weights $\omega_{\text{AJ-GAN}}$ as a Function of Two Characteristics

Notes. The figure illustrates the out-of-sample nonlinear effect on the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN) pricing kernel components $\hat{\omega}^{(T)}_{\text{AJ-GAN}}$ and $\hat{\omega}^{(P)}_{\text{AJ-GAN}}$ (before $\|\omega_t\|_1$ normalization). These effects are illustrated pairwise for the 3 most important characteristics of the permanent component $\hat{\omega}^{(P)}_{\text{AJ-GAN}}$ (left) and the transitory component $\hat{\omega}^{(T)}_{\text{AJ-GAN}}$ (right). Variable importance is based on the saliency analysis in Figure VII. For the permanent component, these characteristics are momentum (r12_2), market capitalization (LME), and sales to operating assets (ATO). For the transitory component, these characteristics are LME, standardized explained variation (SUV), and turnover (Lturnover). All other firm characteristics are kept constant at median levels and hidden states are fixed at zero.



(a) Lturnover × SUV

(b) ST_BV × Bond_r12_3

(c) Lturnover × Resid_Var

(d) ST_BV × LT_Kurt

(e) SUV × Resid_Var

(f) Bond_r12_3 × LT_Kurt

Figure A13

$\text{AJ}^{(\text{bond})}$ Weights $\omega_{\text{AJ}^{(\text{bond})}}$ as a Function of Two Characteristics

Notes. The figure illustrates the out-of-sample nonlinear effect on the Alvarez and Jermann (2005) version of the generative adversarial network in which the transitory component is built only using bond data ($\text{AJ}^{(\text{bond})}$). In particular, it illustrates its pricing kernel components $\hat{\omega}^{(\text{T})}_{\text{AJ}^{(\text{bond})}}$ and $\hat{\omega}^{(\text{P})}_{\text{AJ}^{(\text{bond})}}$ (before $\|\omega_t\|_1$ normalization). The blue regions indicate where the model takes a long position and the red regions indicate short positions. These effects are illustrated pairwise for the 3 most important firm characteristics of the permanent component $\hat{\omega}^{(\text{P})}_{\text{AJ}^{(\text{bond})}}$ (left) and the bond characteristics of the transitory component $\hat{\omega}^{(\text{T})}_{\text{AJ}^{(\text{bond})}}$ (right). Variable importance is based on the saliency analysis in Figure VIII. For the permanent component, these firm characteristics are turnover (Lturnover), standardized unexplained variation (SUV), and residualized variance (Resid_Var). For the transitory components, these bond characteristics are short-term bipower variation (ST_BV), 3-month momentum (Bond_r12_3), and long-term kurtosis (LT_Kurt). All other characteristics are kept constant at median levels and hidden states are fixed at zero.

A4. Additional Model Descriptions

Algorithm 1: Empirical minibatch stochastic gradient descent algorithm of the Chen et al. (2023a) asset pricing Generative Adversarial Network (GAN). Glorot and Bengio (2010) weight initialization across ensemble component varies by a seed s. Biases are initialized at zero and batching is non-stochastic without shuffling observations

Initialization:
$\theta = \left(\theta^{(\text{LSTM})}, \theta^{(\text{FFN})}\right) \leftarrow \left(U^{\text{Glorot}}(s), U^{\text{Glorot}}(s)\right)$ ▷ PK Network

Phase I – Unconditional Estimation

Fix the D unconditional moments $Z = [\mathbf{1}, \mathbf{0}, \ldots, \mathbf{0}]$ ▷ set $R^{(\text{test})} = R^{(\text{basis})}$
for number of epochs N^{Unc} **do** ▷ $N^{\text{Unc}} = 32$
    Set-up Kingma and Ba (2014) Adam optimizer
    for number of batches B^{Unc} **do** ▷ $B^{\text{Unc}} = 5$
        Update the pricing kernel by descending its stochastic gradient:

$$\nabla_\theta \frac{1}{B^{\text{Unc}}} \sum_{j=1}^{B^{\text{Unc}}} \mathcal{L}_j(\omega(\theta)|Z(\varphi), \mathcal{I}_t).$$

Part II – Conditional Estimation

Initialization:
$\varphi = \left(\varphi^{(\text{LSTM})}, \varphi^{(\text{FFN})}\right) \leftarrow (U^{\text{Glorot}}(s), U^{\text{Glorot}}(s))$ ▷ COND Network
for number of iterations N^{GAN} **do** ▷ $N^{\text{GAN}} = 5$
    Set-up Kingma and Ba (2014) Adam optimizer
    for number of epochs N^{COND} **do** ▷ $N^{\text{COND}} = 16$
        for number of batches B^{GAN} **do** ▷ $B^{\text{GAN}} = 5$
            Update the adversary by *ascending* its stochastic gradient:

$$\nabla_\varphi \frac{1}{B^{\text{GAN}}} \sum_{j=1}^{B^{\text{GAN}}} \mathcal{L}_j(Z(\varphi)|\omega(\theta), \mathcal{I}_t).$$

    for number of epochs N^{PK} **do** ▷ $N^{\text{PK}} = 64$
        for number of batches B^{GAN} **do** ▷ $B^{\text{GAN}} = 5$
            Update the pricing kernel by *descending* its stochastic gradient:

$$\nabla_\theta \frac{1}{B^{\text{GAN}}} \sum_{j=1}^{B^{\text{GAN}}} \mathcal{L}_j(\omega(\theta)|Z(\varphi), \mathcal{I}_t).$$

Algorithm 2: Empirical minibatch stochastic gradient descent algorithm of the AJ-GAN asset pricing model. Glorot and Bengio (2010) weight initialization across ensemble component varies by a seed s. Biases are initialized at zero and batching is non-stochastic without shuffling observations. $\xi = 0$ is required in the AJ-GAN and AJ$^{(\text{bond})}$ estimation

Initialization: ▷ $\theta = \{\theta_T, \theta_P\}$

$\theta_T = \left(\theta_T^{(\text{LSTM})}, \theta_T^{(\text{FFN})}\right) \leftarrow \left(U^{\text{Glorot}}(s), U^{\text{Glorot}}(s)\right)$ ▷ Transitory PK

$\theta_P = \left(\theta_P^{(\text{LSTM})}, \theta_P^{(\text{FFN})}\right) \leftarrow \left(U^{\text{Glorot}}(s), U^{\text{Glorot}}(s)\right)$ ▷ Permanent PK

Phase I – Unconditional Estimation

Fix the D unconditional moments $Z = [\mathbf{1}, \mathbf{0}, \ldots, \mathbf{0}]$ ▷ set $R^{(\text{test})} = R^{(\text{basis})}$

for number of epochs N^{Unc} **do** ▷ $N^{\text{Unc}} = 32$

  Set-up Kingma and Ba (2014) Adam optimizer

  for number of batches B^{Unc} **do** ▷ $B^{\text{Unc}} = 5$

    Update the pricing kernel by descending its stochastic gradient:

$$\nabla_{\{\theta_T,\theta_P\}} \frac{1}{B^{\text{Unc}}} \sum_{j=1}^{B^{\text{Unc}}} \left[\mathcal{L}_j^{(\text{unpen})}\left(\omega^{(\text{T})}(\theta_T), \omega^{(\text{P})}(\theta_P) | Z(\varphi), \mathcal{I}_t\right) + \Phi_{\xi,j}\left(\theta_P\right)\right]$$

Part II – Conditional Estimation

Initialization:

$\varphi = \left(\varphi^{(\text{LSTM})}, \varphi^{(\text{FFN})}\right) \leftarrow \left(U^{\text{Glorot}}(s), U^{\text{Glorot}}(s)\right)$ ▷ COND Network

for number of iterations N^{GAN} **do** ▷ $N^{\text{GAN}} = 5$

  Set-up Kingma and Ba (2014) Adam optimizer

  for number of epochs N^{COND} **do** ▷ $N^{\text{COND}} = 16$

    for number of batches B^{GAN} **do** ▷ $B^{\text{GAN}} = 5$

      Update the adversary by *ascending* its stochastic gradient:

$$\nabla_{\varphi} \frac{1}{B^{\text{GAN}}} \sum_{j=1}^{B^{\text{GAN}}} \left[\mathcal{L}_j^{(\text{unpen})}\left(Z(\varphi) | \omega^{(\text{T})}(\theta_T), \omega^{(\text{P})}(\theta_P), \mathcal{I}_t\right) + \Phi_{\xi,j}\left(\theta_P\right)\right]$$

  for number of epochs N^{PK} **do** ▷ $N^{\text{PK}} = 64$

    for number of batches B^{GAN} **do** ▷ $B^{\text{GAN}} = 5$

      Update the pricing kernel by *descending* its stochastic gradient:

$$\nabla_{\{\theta_T,\theta_P\}} \frac{1}{B^{\text{GAN}}} \sum_{j=1}^{B^{\text{GAN}}} \left[\mathcal{L}_j^{(\text{unpen})}\left(\omega^{(\text{T})}(\theta_T), \omega^{(\text{P})}(\theta_P) | Z(\varphi), \mathcal{I}_t\right) + \Phi_{\xi,j}\left(\theta_P\right)\right]$$

Feedforward Neural Network (FFN) with Dropout

A feedforward neural network (FFN) is a broad-type estimator of a general function $h = h_\theta(x)$ that combines linear transformations and element-wise nonlinear operations $\sigma(\cdot)$ called

activation functions. Even simple FFNs with 1 layer are universal approximators. This means that they have the flexibility to approximate any continuous function on compact subsets of a Euclidean space $\mathbb{R}^n$, regardless of the complexity of the function (Hornik et al., 1989). In detail, a single layer FFN of m neurons maps from a d-dimensional vector of input features in the following way:

$$h_\theta(x) = \sigma(\mathbf{W}'x + \mathbf{b}) \tag{25}$$

where the additive constant vector $\mathbf{b} \in \mathbb{R}^m$ is called the bias and $\mathbf{W} \in \mathbb{R}^{m \times d}$ is a matrix of weights. The set of weights and biases constitutes the set of parameters θ_{FFN} to be learned. It is the activation function that allows the FFN to approximate nonlinear patterns. As a special case, when the activation function is the identity function $\sigma = I$, the FFN is reduced to a generalized linear model. The rectified linear unit (ReLU) is the most commonly used activation function and applies the following transformation:

$$\text{ReLU}(x) = \max(x, 0) \tag{26}$$

ReLU can be parallelized efficiently and generally allows for quick convergence, but its gradient saturates at $x < 0$ (Goodfellow et al., 2016).[39] In addition to the ReLU, another common activation function considered in this paper is the scaled exponential linear unit (SELU) of Klambauer et al. (2017):

$$\text{SELU}(x) = \begin{cases} \varrho \cdot x & \text{for } x \geq 0, \\ \varrho \cdot \varsigma \cdot (\exp(x) - 1) & \text{for } x < 0, \end{cases} \tag{27}$$

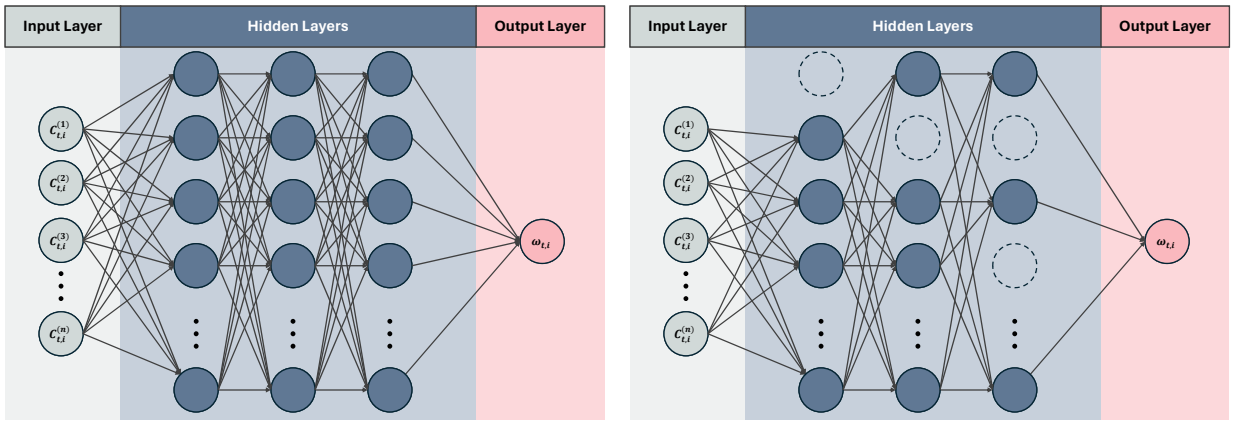
Where the scale parameter $\varrho = 1.67326324$ and $\varsigma = 1.05070098$ are commonly taken from the empirical solutions presented in the original version of Klambauer et al. (2017). These solutions are also adopted in this paper. The linear and non-linear transformations of the FFN can be stacked and repeated for additional layers; see Panel A of Figure A14 for an illustration of a fully dense connected neural network with 3 hidden layers. These transformations propagate through the neural net sequentially for all L layers a_l:

$$a_l = \sigma(\mathbf{W}^{[l]'} a_{l-1} + \mathbf{b}^{[l]}), \qquad \forall l = 1, \ldots, L \tag{28}$$

where $a_0 = x$ is the initial input layer of the data and $h_\theta(x) = \sigma(\mathbf{W}^{[L]'} a_{L-1} + \mathbf{b}^{[L]})$ is

[39] This is also called the *Dying ReLU problem* Which is solved by using Leaky ReLU. Other common activation functions include Sigmoid, Tanh, Softplus, Softmax.

the final output layer of the model. All layers a_l for which $1 < l < L$ are called hidden layers. The number of hidden layers is commonly used as a proxy for the "depth" of a neural network. The general rule of thumb is that when the network has at least 3 hidden layers, you are in a deep learning environment (Goodfellow et al., 2016). The benefit of adding depth is that hidden layers can perform additional iterations of feature extraction. A supervised learning setup means that you have pairs $\{y_i, \boldsymbol{x}_i\}$ of outcomes y_i and features $\boldsymbol{x}_i$ for $i = 1, \ldots, N$ examples. The performance of the FFN is measured with a loss function $\mathcal{L}(y, h_\theta(\boldsymbol{x}))$, where $\hat{y} = h_\theta(\boldsymbol{x})$. The researcher has the flexibility to choose which loss function to use, but generally the loss is a form of distance metric between outcomes y and predictions $\hat{y}$. Common loss regression-type loss functions include the mean squared error (MSE) and the mean absolute error (MAE). FFNs are trained using the backpropagation algorithm, which involves calculating the gradient of the loss function with respect to all weights and biases; $\nabla_{\theta_{\text{FFN}}}(\mathcal{L})$. In practice, this involves using the chain rule iteratively, as the calculations are traced backward through the neural network until you reach the input layer.



Panel A. FFN Without Dropout

Panel B. FFN With Dropout

Figure A14

Feedforward Neural Network (FFN) and Dropout

Notes. Panel A illustrates a fully dense feedforward neural network (FFN) model with three hidden layers. To illustrate, the FFN may take firm characteristics $C_{t,i} \in I_{t,i}$ and transform it into a PK weight $\omega_{t,i}$. Panel B illustrates the same neural network but with a dropout mask applied. Note that it is typically considered bad practice to apply dropout just prior to the output layer as there is over-regularization and little to learn by the model. Therefore, dropout is not applied on the penultimate layer of any neural network in this paper. The figure is adapted from (Goodfellow et al., 2016).

A common form of regularization on neural networks is dropout. The purpose of dropout is to reduce overfitting by introducing noise to the model architecture. This is achieved by applying a stochastic mask on the nodes of the neural network during training. This mask means that the researcher randomly removes a certain number of nodes from the neural

network each training iteration. Crucially, it is not necessarily the same nodes that are removed in each iteration. The degree of regularization is measured using the dropout rate. A higher dropout rate means that a larger portion of the nodes are dropped each iteration. Dropping nodes achieves 2 goals; first it reduces the capacity of the network during training. This means that the model has less capacity and therefore it will take more time to simply memorize training examples. Due to reduced capacity, a caveat of dropout is that researchers will usually have to run their estimation procedure for a longer time until the model reaches convergence. Second, it introduces artificial noise in the estimation which makes it more difficult to overfit on the data. Instead, the model should be able to find more generalizable patterns in the data. In practice, the optimal dropout rate is usually chosen from a grid of candidate values using a data-driven validation procedure, such as cross-validation.[40] See panel B of Figure A14 for an illustration of a feedforward network with dropout.

Recurrent Neural Networks and Long Short-Term Memory

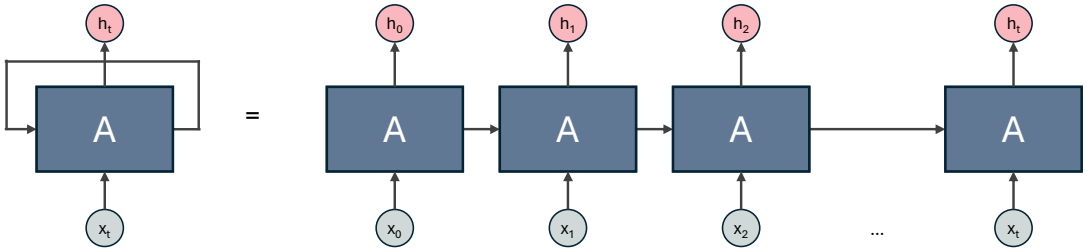
In the most general setting, the information set $\mathcal{I}_t$ of an investor expands as time passes and new information is generated. It is likely that both current and past characteristics are likely to be informative about the risk-reward relationship of an asset. Naturally, more recent data contain more relevant information than older data. Econometric models, such as ARMA-GARCH type models, can be used to capture such long-term dependencies while prioritizing more recent data. In the machine learning literature, these long-term dependencies are usually modeled using a class of neural networks called recurrent neural networks (RNNs). The input of an RNN at time t is an observable sequence of vector-valued inputs $\{x_0, \ldots, x_t\}$ (Goodfellow et al., 2016). The RNN transforms this sequence to an output h_t using a neural network module A.[41] The architecture of the RNN can be thought of as a normal neural network model A combined an added feedback loop. Illustratively, this feedback loop can be unrolled; see Panel A of Figure A15. That is, the RNN can be construed as several replicas of the same neural network module A grouped in a sequence. This architecture of the RNN makes it particularly adept at handling sequence problems such as language modeling, speech recognition, or time series dynamics. The training of RNNs is done by a variation of the ordinary back-propagation algorithm called back-propagation through time (BPTT). Although RNNs should be able to capture all long-term dependencies in theory, the BPTT algorithm is known to break down frequently when considering very long sequences of data. This is because of the multiplicative nature of the chain rule. In particular, the sensitivity of the objective function to far-away lags is usually small. When these sensitivities are

[40] Both Gu et al. (2020) and Chen et al. (2023a) use a dropout rate of 0.05

[41] The notation for h_t is chosen to denote a "hidden state" h at time t in the typical asset pricing context

multiplied together, the overall gradient vanishes.[42] Alternatively, a few large sensitivities multiplied together leads to an exploding gradient problem. A very popular variant of the basic RNN model is the long short-term memory (LSTM) network. This was introduced by Hochreiter and Schmidhuber (1997) as a solution to the problem of vanishing and exploding gradients. The LSTM share the chained modular structure of the RNN, but each module is constructed very differently, see panel B in Figure A15.

Panel A. Stacked and Unrolled Recurrent Neural Network (RNN)



Panel B. Long Short-Term Memory (LSTM) Cells

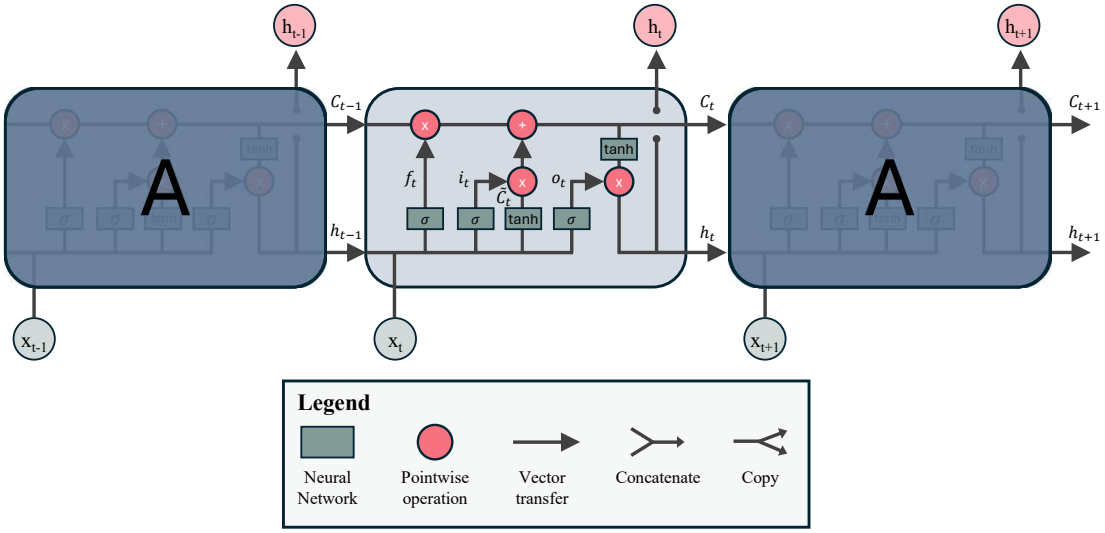


Figure A15
Recurrent Neural Network (RNN) and Long-Short Term Memory (LSTM) Cells

Notes. Panel A illustrates a Recurrent Neural Network (RNN) in stacked form (left) and in unrolled form (right). See text for details on the RNN. Panel B illustrates the architecture of a Long Short-Term Memory (LSTM) model. Merging lines illustrates concatenations and separating lines illustrates duplications. End-point of arrows illustrates the transfer of a vector. See text for detailed description of all components of the LSTM cell. Illustrations are adapted from Goodfellow et al. (2016)

The upper horizontal line in the LSTM cell takes as input a previous cell state C_{t-1} and applies two operations to it to produce the current cell state C_t (Goodfellow et al., 2016). The

[42]This is called the vanishing gradient problem (see e.g. Hochreiter, 1991 and Bengio et al., 1994).

first operation is a point-wise multiplication with the forget gate layer f_t, which is composed of a neural net layer with a sigmoid activation function $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$. This means that, depending on the previous state h_t and the current covariates x_t, the old cell state is multiplied by a number in $[0, 1]$ which regulates how much information is sent through the gate. The trainable parameters are the weights and biases of the forget gate W_f and b_f, respectively. The second operation on the cell state is a combination of an input gate layer $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$ and a candidate cell state $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$. Their element-wise product $i_t \cdot \tilde{C}_t$ is ultimately added to the cell state. The intuition is that this operation corresponds to adding new information to the cell state. The candidate cell $\tilde{C}_t$ corresponds to the new information, and the input gate layer i_t corresponds to a fraction in $[0, 1]$ of how much of this information is added to the cell state. This is how the cell state is updated to include new information x_t. Therefore, the complete update comes from: $C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$. In this case, the trainable parameters are the pair of weights W_i and W_C and biases b_i and b_C. In addition to the cell state, the LSTM produces a filtered hidden state variable $h_t = o_t \cdot \tanh(C_t)$ in which the output gate layer o_t is a function of the prior hidden state h_{t-1} and the contemporaneous covariates x_t, that is; $o_t \cdot \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$. The trainable parameters are W_o and b_o. In total, this means that the baseline LSTM cell has a set of 4 weight parameters $\{W_f, W_i, W_C, W_o\}$ and a set of 4 biases $\{b_f, b_i, b_C, b_o\}$ to be trained. This type of baseline LSTM cell is the one adopted in the GAN model of Chen et al. (2023a).

Elastic Net

The elastic net of Zou and Hastie (2005) is a linear regression model that performs both coefficient shrinkage and selection. In detail, the elastic net adds a l_1-penalty to the regression and a l_2-penalty to a typical regression model. That is, the elastic net merges the lasso model of Tibshirani (1996) with the classical ridge regression of Hoerl and Kennard (1970). Given labels $\mathbf{y}$ and features $\mathbf{X}$, the elastic net solves the following optimization problem:

$$\boldsymbol{\beta}^* \in \arg\min_{\boldsymbol{\beta}} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \frac{\alpha}{2}(1 - \lambda)||\boldsymbol{\beta}||_2^2 + \alpha\lambda||\boldsymbol{\beta}||_1 \tag{29}$$

where $\boldsymbol{\beta}$ is a matrix of coefficients that includes an intercept. The hyperparameter α is a constant measuring the overall shrinkage to apply, and the hyperparameter λ is the mixing parameter that determines the ratio of l_1 and l_2 shrinkage. These hyperparameters are usually selected based on informed research priors or by using a data-driven validation procedure, e.g., cross-validation. The total penalty term is called the elastic net penalty. The contribution of the Lasso regularization ($||\boldsymbol{\beta}||_1$) is to set the coefficients of the least

informative predictors to zero. This can lead to a high degree of sparsity. On the other hand, Ridge regularization ($||\boldsymbol{\beta}||_2^2$) contributes by shrinking the coefficients of less informative predictors towards 0 without completely regularizing them away. Note that when $\alpha = 0$ the EN simplifies to the standard Lasso model. In contrast, when $\alpha = 1$ the EN is reduced to the Ridge model. The optimization problem in Equation (29) is usually solved using the Least-Angle Regression Elastic Net (LARS-EN) algorithm of Zou and Hastie (2005).

A5. Supplementary Data Description

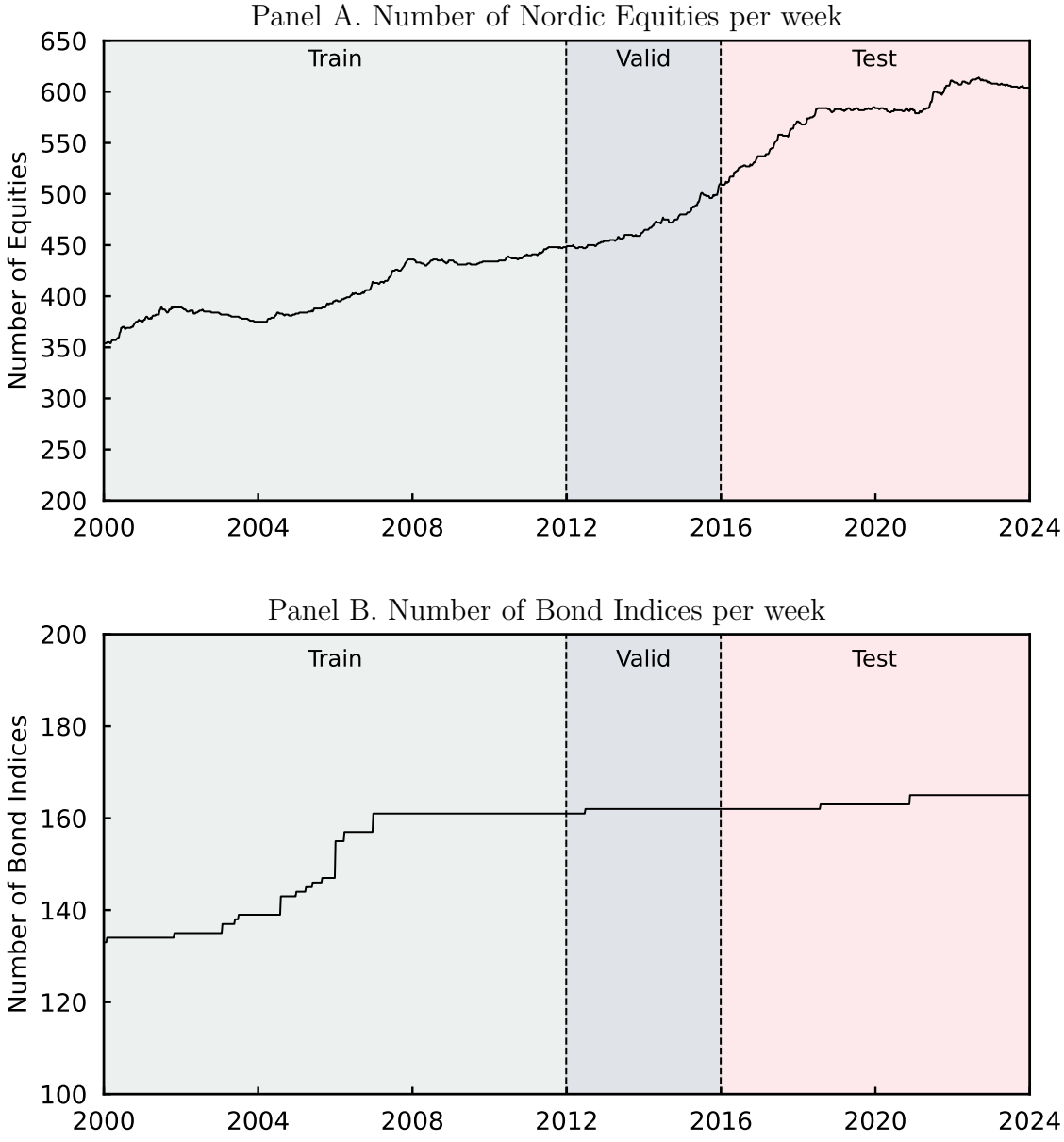


Figure A16

Number of Assets in the Dataset by Week

Notes. The figures illustrate the time-series component of the unbalanced panel used in this study. The solid line corresponds to the number of assets included in the sample at each week t. Panel A illustrates the number of Nordic equities included, and panel B illustrates the number of bond indices included. The sampling process is detailed in Section III. The shaded regions correspond to the separation of the overall sample into the training set (2000/01/04 - 2011/12/20), validation set (2011/12/27 - 2015/12/29), and test set (2015/12/29 - 2023/12/19).

Table A7: Part 1

Detailed Overview of Firm-Characteristics

	Acronym	Name	Category	Definition (Acronym) [Eikon Code]
(1)	AC	Accrual	Intangibles	YoY change in operating working capital (OWC) per share [NOSH] divided by book equity (defined in BEME) per share. OWC is total current assets [WC02201/A] minus cash & cash equivalents (C&CE) [WC02001/A] minus total current liabilities [WC03101/A] minus short-term debt [WC03051/A] minus taxes payable [WC03063/A].
(2)	OA	Operating Accrual	Intangibles	YoY change in OWC (defined in AC) minus depreciation & amortization [WC01151/A] divided by 52-week lag of total assets (AT).
(3)	OL	Operating Leverage	Intangibles	Sum of cost of goods sold (COGS) [WC01051/A] and selling, general, and administrative expenses (SG&A) [WCO1101/A] over total assets (AT)
(4)	PCM	Pice-to-Cost Margin	Intangibles	Sales [WC01001/A] less costs of goods sold (COGS), scaled by sales
(5)	AT	Assets	Frictions	Latest available total assets [WC02999/A]
(6)	Beta	Short-term beta	Frictions	Regression coefficient of daily excess returns in a 180-trading day rolling-window regression onto the daily excess returns of the market-weighted portfolio of all stocks. Regression includes a constant.
(7)	IdioVol	Idiosyncratic Volatility	Frictions	Rolling 25-week volatility of residuals from a 252 trading-day rolling-window regression of daily excess returns on the daily returns on the market-weighted portfolio of all stocks and the returns on the two 10-1 decile-sorted portfolios based on size (LME) and value (BEME). Regression includes a constant.
(8)	LME	Market Capitaliza-tion	Frictions	Prior week total market capitalization [MV]
(9)	Lturnover	Turnover	Frictions	Sum of daily trading volume [VO] for the last month scaled by shares outstanding [NOSH]
(10)	MktBeta	Long-Term Beta	Frictions	Regression coefficient of daily excess returns from a 250-week rolling-window regression on excess return of the market-weighted portfolio of all stocks. Regression includes a constant.
(11)	Rel2High	Closeness to High	Frictions	The ratio of the current value of the stock's total return-index to a 250-week rolling-window maximum.
(12)	Resid_Var	Residual Variance	Frictions	Variance of residuals from a 90-day rolling-window regression of daily excess returns to the daily excess returns of the market-weighted portfolio of all stocks and the two 10-1 decile-sorted portfolios based on size (LME) and value (BEME). Regression includes a constant.
(13)	SUV	Standard Unex-plained Variation	Frictions	Four-week rolling sum of daily prediction errors of actual volume versus fitted volume. The predicted volume comes from a predictive predictive regression of tomorrows volume onto the absolute values of positive returns and negative returns. Regression includes a constant and the sum is scaled by the volatility of the residuals over the same period.
(14)	Spread	Bid-Ask-Spread	Frictions	Four-week rolling average of the difference of daily ask prices [PA] and bid prices [PB]. The bid-ask spread scaled by the mid-point price [P].
(15)	Variance	Variance	Frictions	70-day rolling-window variance of daily excess returns.

Notes: ll firm-level data used to construct these characteristics are retrieved from Refinitiv Eikon, with acronyms in parentheses and Eikon codes in brackets. In order to follow Chen et al. (2023a) as closely as possible, the variables are defined as closely as possible, with necessary adjustments for the weekly frequency and the available data. Where possible, income statement and cash flow metrics are adjusted to latest available trailing-twelve-months (TTM) at each point in time. The interim items have an Eikon code with suffix "A". Balance sheet variables are always the latest available figure from interim or annual reports. See Chen et al. (2023b) for literary references relating to each firm-characteristic.

Table A7: (Continued) Part 2
Detailed Overview of Firm-Characteristics

	Acronym	Name	Category	Definition (Acronym) [Eikon Code]
(16)	LT_Rev	Long-Term Reversal	Returns	Cumulative returns from 260 weeks (5 years) before the current date to 52 weeks (1 year) prior to the current date.
(17)	ST_Rev	Short-Term Reversal	Returns	Prior week return
(18)	r12_2	Momentum	Returns	Cumulative returns from 52 weeks (12 months) prior to the current date to 8 weeks (2 months) prior to the current date.
(19)	r12_7	Intermediate Momentum	Returns	Cumulative returns from 52 weeks (12 months) prior to the current date to 28 weeks (7 months) prior to the current date.
(20)	r2_1	Short-Term Momentum	Returns	Cumulative returns from 8 weeks (2 months) prior to the current date to 4 weeks (1 month) prior to the current date
(21)	r36_13	Long-Term Momentum	Returns	Cumulative returns from 156 weeks (36 months) prior to the current date to 58 weeks (13 year) prior to the current date.
(22)	Investment	Investment	Investment	YoY change in total assets (AT) divided by the 52-week lag of total assets.
(23)	NOA	Net Operating Assets	Investment	Net of operating assets and liabilities, scaled by the 52-week lag of total assets (AT). Operating assets is total assets less C&CE, minus investment and other advances [WC02255/A]. Operating liabilities is total assets minus current debt [WC03051/A] minus long-term debt [WC03251/A] minus boojk values of non-controlling interests (NCI) [WC03426/A], minus book value of preferred stock (PS) [WC03451/A] minus common equity (CEQ) [WC03501/A].
(24)	NSI	Net Share Issues	Investment	Lagged YoY log-difference in shares outstanding [NOSH].
(25)	DPI2A	Change in PP&E	Investment	YoY changes in property, plants, and equipment (PP&E) [WC02501/A] and inventory [WC02101/A] over total assets (AT)
(26)	ATO	Sales to operating assets	Profitability	Sales [WC01001/A] to over last weeks net operating assets (defined in NOA).
(27)	CTO	Capital Turnover	Profitability	Sales [WC01001/A] to last weeks total assets (AT)
(28)	D2A	Capital Intensity	Profitability	Ratio of depreciation and amortization [WC01151/A] to total assets (AT)
(29)	FC2Y	Fixed Cost	Profitability	Sum of expenses relating to SG&A [WC01101/A] and research & development (R&D) [WC01201/A], scaled by sales [WC01001/A]
(30)	OP	Operating Profit	Profitability	Sales [WC01001/A] minus the sum of COGS [WC01051/A], interest expenses [WC01075/A] and SG&A [WC01101/A], then scaled by book value equity (defined in BEME)
(31)	PM	Profit Margin	Profitability	Operating profit [WC01250/A] to sales [WC01001/A]

Notes: ll firm-level data used to construct these characteristics are retrieved from Refinitiv Eikon, with acronyms in parentheses and Eikon codes in brackets. In order to follow Chen et al. (2023a) as closely as possible, the variables are defined as closely as possible, with necessary adjustments for the weekly frequency and the available data. Where possible, income statement and cash flow metrics are adjusted to latest available trailing-twelve-months (TTM) at each point in time. The interim items have an Eikon code with suffix "A". Balance sheet variables are always the latest available figure from interim or annual reports. See Chen et al. (2023b) for literary references relating to each firm-characteristic.

Table A7: (Continued) Part 3
Detailed Overview of Firm-Characteristics

	Acronym	Name	Category	Definition (Acronym) [Eikon Code]
(32)	PROF	Profitability	Profitability	Gross profit [WC0110/A] to book value of equity (defined in BEME)
(33)	RNA	Return on NOA	Profitability	Operating income [WC01250/A] to the 1-week lag of net operating assets (defined in NOA)
(34)	ROA	Return on Assets	Profitability	Adjusted net income [WC01551/A] to prior week total assets (AT). Adjusted net income is a normalized net income metric excluding extraordinary items.
(35)	ROE	Return on Equity	Profitability	Adjusted net income [WC01551/A] to prior week book-value of equity (defined in BEME)
(36)	SGA2S	SG&A to Sales	Value	Selling, general and administrative expenses (SG&A) [WC01101/A] to sales [WC01011/A]
(37)	A2ME	Assets to Equity	Value	Total assets (AT) over market capitalization [MV]
(38)	BEME	Book-to-Market	Value	Ratio of book value of equity to the 1-week lag of market value of equity [MV]. Book value is the sum of shareholder equity [WC03995/A] and deferred taxes [WC04101] minus book value of preferred stock [WC03451/A]. If shareholder equity is missing, it's proxied by the sum of common equity [WC03501/A] and preferred stock [WC03451/A]. If still missing, shareholders equity is replaced by the difference of total assets (AT) and total liabilities [WC03351/A].
(39)	C	Cash to Assets	Value	C&CE [WC02001/A] to total assets (AT)
(40)	CF	FCF to equity	Value	Free cash flow (FCF) to book value of equity (defined in BEME). Free cash flow is defined as adjusted net income [WC01551/A] plus depreciation and amortization [WC01151/1] less change in OWC (defined in AC) capital expenditures [WC04601/A]
(41)	CF2P	Cashflow to Price	Value	Cashflow over the 1-week lag of market capitalization [MV]. Cashflow is defined as income adjusted net income [WC01551/A] plus depreciation and amortization [WC01151/A] plus deferred income taxes [WC04199/A].
(42)	DY	Dividend Yield	Value	Dividend yield [DY].
(43)	E2P	Earnings to Price	Value	Adjusted net income [WC01551/A] to market value of equity [MV]
(44)	Lev	Leverage	Value	Long-term interest-bearing debt [WC03251/A] and short-term interest-bearing debt [WC03051/A] over long-term and short-term interest bearing debt and shareholders equity [WC03501/A]
(45)	Q	Tobin's Q	Value	Total assets (AT) plus market capitalization [MV] less C&CE [WC02001/A] and taxes payable [WC03063/A], all scaled by total assets (AT)
(46)	S2P	Sales-to-Price	Value	Sales [WC01001/A] to market capitalization [MV]

Notes: All firm-level data used to construct these characteristics are retrieved from Refinitiv Eikon, with acronyms in parentheses and Eikon codes in brackets. In order to follow Chen et al. (2023a) as closely as possible, the variables are defined as closely as possible to their paper, with necessary adjustments made due to the weekly frequency, data availability, and the Nordic setting. Where possible, income statement and cash flow metrics are adjusted to latest available trailing-twelve-months (TTM) at each point in time. The interim items have an Eikon code with suffix "A". Balance sheet variables are always the latest available figure from interim or annual reports. See Chen et al. (2023b) for literary references relating to each firm characteristic.

Table A8

Detailed Overview of Macroeconomic Firm Variables

	Underlying Characteristic		Transformations			Underlying Characteristic		Transformations	
	Acronym	Name	Aggregation	tCode		Acronym	Name	Aggregation	tCode
		Intangibles					**Investment**		
(1)	AC	Accrual	Median	[2]	(22)	Investment	Investment	Median	[2]
(2)	OA	Operating Accrual	Median	[2]	(23)	NOA	Net Operating Assets	Median	[3]
(3)	OL	Operating Leverage	Median	[3]	(24)	NSI	Net Share Issues	Mean	[1]
(4)	PCM	Price-to-Cost	Median	[3]	(25)	DPI2A	Change in PP&E	Median	[2]
		Frictions					**Profitability**		
(5)	AT	Total Assets	Median	[4]	(26)	ATO	Sales to OA	Median	[3]
(6)	Beta	Short-Term Beta	Median	[1]	(27)	CTO	Capital Turnover	Median	[3]
(7)	IdioVol	Idiosyncratic Volatility	Median	[3]	(28)	D2A	Capital Intensity	Median	[3]
(8)	LME	Market Capitalization	Median	[4]	(29)	FC2Y	Fixed Cost to Sales	Median	[3]
(9)	Lturnover	Turnover	Median	[3]	(30)	OP	Operating Profit	Median	[3]
(10)	MktBeta	Long-Term Beta	Median	[1]	(31)	PM	Profit Margin	Median	[3]
(11)	Rel2High	Relative to High	Median	[3]	(32)	PROF	Profitability	Median	[3]
(12)	Resid_Var	Residual Variance	Median	[3]	(33)	RNA	Return of NOA	Median	[3]
(13)	SUV	Unexplained Variation	Median	[1]	(34)	ROA	Return on Assets	Median	[3]
(14)	Spread	Bid-Ask-Spread	Median	[3]	(35)	ROE	Return on Equity	Median	[3]
(15)	Variance	Variance	Median	[3]	(36)	SGA2S	SG&A to Sales	Median	[3]
		Returns					**Value**		
(16)	LT_Rev	Long-Term Reversal	Median	[2]	(37)	A2ME	Assets to Market Cap.	Median	[3]
(17)	ST_Rev	Short-Term Reversal	Median	[2]	(38)	BEME	Book-to-Market	Median	[3]
(18)	r12_2	Momentum	Median	[2]	(39)	C	Cash to Assets	Median	[3]
(19)	r12_7	Intermediate Momentum	Median	[2]	(40)	CF	FCF to Equity	Median	[2]
(20)	r2_1	Short-Term Momentum	Median	[2]	(41)	CF2P	Cashflow to Price	Median	[3]
(21)	r36_13	Long-Term Momentum	Median	[2]	(42)	DY	Dividend Yield	Median	[2]
					(43)	E2P	Earnings to Price	Median	[3]
					(44)	Lev	Leverage	Median	[3]
					(45)	Q	Tobin's Q	Median	[3]
					(46)	S2P	Sales to Price	Median	[3]

Notes: This table details the transformations used to construct the macroeconomic firm characteristics. These transformations are adopted from Chen et al. (2023a) but adapted where necessary. These authors also include additional US-specific macroeconomic variables that are not included in this study. The reason for excluding additional macroeconomic variables is to keep the estimation procedure simple due to the computational intensity and training time of an LSTM cell. The transformations remove the most apparent nonstationarities and keep the macroeconomic variables on a similar scale. All underlying macroeconomic variables at time t are taken as cross-sectional medians of the non-imputed firm characteristics at any time t. For a detailed overview of the data sources and the construction of the underlying firm characteristics, see Table A7. Transformation codes (tCode) are as follows: [1] x_t; [2] Δx_t; [3] $\Delta \log(x_t)$; [4] $\Delta^2 \log(x_t)$.

Table A9: Part 1
Detailed Overview of Bond-Characteristics

	Acronym	Name	Definition	Reference
(1)	AMIHUD	Unscaled Amihud	Rolling 3-month average of the weekly unscaled Amihud illiquidity measure. The unscaled illiquidity measure in a week is the absolute return of that week.	Amihud (2002)
(2)	AMOUT	Amount Outstanding	Number of bond issues outstanding	Houweling et al. (2005)
(3)	Bond_LT_Rev	Long-term reversal	Return between 3 years prior to the current date to 1 year prior to the current date	Bali et al. (2021)
(4)	Bond_ST_Rev	Short-term reversal	The most recent lagged return multiplied by -1	Bai et al. (2019)
(5)	Bond_r12_3	3M momentum	Cumulative returns from 52 weeks (12 months) prior to the current date to 12 weeks (3 months) prior to the current date	Jostova et al. (2013)
(6)	Bond_r12_6	6M momentum	Cumulative returns from 52 weeks (12 months) prior to the current date to 24 weeks (6 months) prior to the current date	Jostova et al. (2013)
(7)	Bond_r12_9	9M momentum	Cumulative returns from 52 weeks (12 months) prior to the current date to 36 weeks (9 months) prior to the current date	Jostova et al. (2013)
(8)	CBeta	Short-Term Credit Beta	12 month rolling beta to the value-weighted portfolio of bond indices	Fama and French (1993)
(9)	CMktBeta	Long-Term Credit Beta	Regression coefficient of daily excess returns from a 250-week (5-year) rolling-window regression on excess return of the market-weighted portfolio of all bond indices.	Fama and French (1993)
(10)	CSpread	Credit Spread	Weighted-average yield to maturity less yield on shortest-term treasuries in the country of the bond index	Israel et al. (2018)
(11)	ES10	10% Expected Shortfall	The rolling 36-week average of the 4 lowest returns, multiplied by -1	Bai et al. (2019)
(12)	ES5	10% Expected Shortfall	The rolling 36-week average of the 2 lowest returns, multiplied by -1	Bai et al. (2019)
(13)	Jump	Jump Variation	Rolling 52-week jump variation: $\mathrm{JV}_{t,i} = \sum_{\tau=t-52}^{t} \left([R^{e,(b)}_{\tau+1,i}]^2 - \sigma^2_{\tau+1,i} \right)^+$, where $\sigma^2_{t+1,i}$ is the 52-week rolling variance of bond returns $R^{e,(b)}_{t+1,i}$ for bond i	Barndorff-Nielsen and Neil (2006)
(14)	LME	Market Capitalization	Market capitalization of constituent bonds	Gebhardt et al. (2005)
(15)	LT_BV	Long-Term Bipower Variation	156-week (3-year) rolling Bipower Variation: $\mathrm{BV}_{t,i} = \frac{\pi}{2} \cdot \frac{1}{155} \sum_{\tau=t-155}^{t} \lvert R^{e,(b)}_{\tau,i} \rvert \cdot \lvert R^{e,(b)}_{\tau-1,i} \rvert$	Barndorff-Nielsen and Neil (2004)

Notes: This table details the construction of the bond characteristics. All data is retrieved from Refinitiv Eikon. The final column provides a reference to the academic paper from which the listed factor or metric is adapted. As most of the the referenced literature works with monthly returns (or high-frequency data) some necessary adaptations are made.

Table A9: (Continued) Part 2
Detailed Overview of Bond-Characteristics

	Acronym	Name	Definition [Eikon Code]	Reference
(16)	LT_Bond_Vol	Long-Term Volatility	Rolling 156-week bond return volatility	Bai et al. (2019)
(17)	LT_Kurt	Long-Term Kurtosis	156-week (3 year) bond return kurtosis	Bai et al. (2019)
(18)	LT_Range	Long-Term Realized Range	156-week (3-year) rolling and scaled realized range: $\text{RR}_{t,i} = \frac{\left(P_{t,i}^{\text{high}} - P_{t,i}^{\text{low}}\right)^2}{P_{t,i}^2}$. $P_{t,i}^{\text{high}}$ and $P_{t,i}^{\text{high}}$ denote the 156-week rolling high and low prices of bond i.	Parkinson (1980)
(19)	LT_Skew	Long-Term Skewness	Rolling 156-week return skewness	Bai et al. (2019)
(20)	Life	Life-to-maturity	The weighted-average maturity of the constituent bond instruments in the bond index	Houweling et al. (2005)
(21)	MD	Duration	Weighted-average modified Macaulay duration of index constituents	Israel et al. (2018)
(22)	ST_BV	Short-Term Bipower Variation	36-week rolling Bipower Variation: $\text{BV}_{t,i} = \frac{\pi}{2} \cdot \frac{1}{35} \sum_{\tau=t-36}^{t} \lvert R_{\tau,i}^{e,(b)} \rvert \cdot \lvert R_{\tau-1,i}^{e,(b)} \rvert$	Barndorff-Nielsen and Neil (2004)
(23)	ST_Bond_Vol	Short-Term Volatility	Rolling 36-week bond return volatility	Bai et al. (2019)
(24)	ST_Kurt	Short-Term Kurtosis	Rolling 36 week return kurtosis	Bai et al. (2019)
(25)	ST_Range	Short-Term Realized Range	36-week rolling and scaled realized range: $\text{RR}_{t,i} = \frac{\left(P_{t,i}^{\text{high}} - P_{t,i}^{\text{low}}\right)^2}{P_{t,i}^2}$. $P_{t,i}^{\text{high}}$ and $P_{t,i}^{\text{high}}$ denote the 36-week rolling high and low prices of bond i.	Parkinson (1980)
(26)	ST_Skew	Short-Term Skewness	Rolling 36-week return skewness	Bai et al. (2019)
(27)	TQ	Tripower Quarticity	156-week rolling tripower quarticity: $\text{TQ}_{t,i} = \frac{2\sqrt{2}}{3} \cdot \frac{1}{154} \sum_{\tau=t-154}^{t} \lvert R_{\tau+1,i}^{e,(b)} \rvert^{4/3} \cdot \lvert R_{\tau,i}^{e,(b)} \rvert^{4/3} \cdot \lvert R_{\tau-1,i}^{e,(b)} \rvert^{4/3}$	Barndorff-Nielsen and Neil (2004)
(28)	VaR10	10% Value-at-Risk	The 4^{th} lowest return over a rolling 36 week window, multiplied by -1	Bai et al. (2019)
(29)	VaR5	5% Value-at-Risk	The 2^{nd} lowest return over a rolling 36 week window, multiplied by -1	Bai et al. (2019)
(30)	YTM	Yield-to-maturity	Weighted-average promised yield-to-maturity	Gebhardt et al. (2005)

Notes: This table details the construction of the bond characteristics. All data is retrieved from Refinitiv Eikon. The final column provides a reference to the academic paper from which the listed factor or metric is adapted. As most of the the referenced literature works with monthly returns (or high-frequency data) some necessary adaptations are made.

Table A10: Part 1
Detailed Overview of Bond-Indices

	Code	Country	Description		Code	Country	Description
(1)	ASDGVAL	Sweden	DS gov't all mats	(49)	SDOMTBD	Sweden	OMX t-bond benchmark
(2)	ASDGVG1	Sweden	DS all gov't 1-3Y	(50)	SDOMTBL	Sweden	OMX t-bill and corp
(3)	ASDGVG2	Sweden	DS all gov't 3-5Y	(51)	SDOMTOT	Sweden	OMX all bonds
(4)	ASDGVG3	Sweden	DS all gov't 5-7Y	(52)	SPFISEI	Sweden	S&P inflation-linked
(5)	ASDGVG4	Sweden	DS all gov't 7-10Y	(53)	SPSEGQR	Sweden	S&P quasi gov't
(6)	ASDGVG5	Sweden	DS all gov't 10Y+	(54)	SPSEI1R	Sweden	S&P IG corp 1Y+
(7)	BMSD02Y	Sweden	DS gov't 2Y	(55)	SPSEI3A	Sweden	S&P all AAA corp
(8)	BMSD03Y	Sweden	DS gov't 3Y	(56)	SPSEI3B	Sweden	S&P all BBB corp
(9)	BMSD05Y	Sweden	DS gov't 5Y	(57)	SPSEI3R	Sweden	S&P IG corp 0-3Y
(10)	BMSD07Y	Sweden	DS gov't 7Y	(58)	SPSEI5R	Sweden	S&P IG corp 3-5Y
(11)	BMSD10Y	Sweden	DS gov't 10Y	(59)	SPSEI7R	Sweden	S&P IG corp 5-7Y
(12)	HMSD13Y	Sweden	SHB gov't 1-3Y	(60)	SPSEICR	Sweden	S&P all IG corp
(13)	HMSD25Y	Sweden	SHB gov't 2.5D	(61)	SPSEIDR	Sweden	S&P IG corp 7-10Y
(14)	HMSD35Y	Sweden	SHB gov't 3-5Y	(62)	SPSEILR	Sweden	S&P IG corp 10Y+
(15)	HMSDALL	Sweden	SHB gov't all mats	(63)	SPSEQ2A	Sweden	S&P AA quasi gov't
(16)	HMSDD3Y	Sweden	SHB gov't 3D	(64)	SPSEQ3A	Sweden	S&P AAA quasi gov't
(17)	HMSDD4Y	Sweden	SHB gov't 4D	(65)	SPSFISV	Sweden	S&P sovereign bonds
(18)	HMSDD5Y	Sweden	SHB gov't 5D	(66)	SPSGQ1R	Sweden	S&P quasi gov't 1Y+
(19)	HMSDD6Y	Sweden	SHB gov't 6D	(67)	SPSGQ3R	Sweden	S&P quasi gov't 0-3Y
(20)	HMSDMGE	Sweden	SHB all mge	(68)	SPSGQ5R	Sweden	S&P quasi gov't 1-5Y
(21)	HMSDO10	Sweden	SHB gov't 10Y+	(69)	SPSGQD1	Sweden	S&P quasi gov't 10Y+
(22)	HMSDO5Y	Sweden	SHB gov't 5-10Y	(70)	SPSGQD5	Sweden	S&P quasi gov't 3-5Y
(23)	HMSDTOT	Sweden	SHB all	(71)	SPSGQDR	Sweden	S&P quasi gov't 5-10Y
(24)	HMSLALL	Sweden	SHB all index-linked	(72)	SPSI15R	Sweden	S&P IG corp 1-5Y
(25)	HMSM13Y	Sweden	SHB mge 1-3Y	(73)	SPSI50R	Sweden	S&P IG corp 5-10Y
(26)	HMSM35Y	Sweden	SHB mge 3-5Y	(74)	SPSIA1R	Sweden	S&P A IG corp
(27)	HMSMO5Y	Sweden	SHB mge 5-10Y	(75)	SPSIA2R	Sweden	S&P AA IG corp
(28)	HMTMD05	Sweden	SHB treasury mge 5D	(76)	SSVX90T	Sweden	OMX t-bill 90d
(29)	HMTS13M	Sweden	SHB treasury 1-3M	(77)	TSDGVAL	Sweden	All gov't tracker
(30)	HMTSALL	Sweden	SHB all treasury	(78)	TSDGVG1	Sweden	gov't 1-3Y tracker
(31)	OMRX180	Sweden	OMX t-bill 180d	(79)	TSDGVG2	Sweden	gov't 3-5Y tracker
(32)	OMRXA13	Sweden	OMX all bond 1-3Y	(80)	TSDGVG3	Sweden	gov't 3-5Y tracker
(33)	OMRXA5Y	Sweden	OMX all bond 5Y+	(81)	TSDGVG4	Sweden	gov't 5-7Y tracker
(34)	OMRXB13	Sweden	OMX t-bond 1-3Y	(82)	TSDGVG5	Sweden	gov't 10Y+ tracker
(35)	OMRXB35	Sweden	OMX t-bond 3-5Y	(83)	SPFIISK	Iceland	S&P gov't all mats
(36)	OMRXBAL	Sweden	OMX all t-bond	(84)	SPFIISU	Iceland	S&P all gov't
(37)	OMRXM13	Sweden	OMX mge 1-3Y	(85)	AFNGVAL	Finland	DS gov't all mats
(38)	OMRXM35	Sweden	OMX mge 3-5Y	(86)	BMFN03Y	Finland	DS gov't 3Y
(39)	OMRXM5Y	Sweden	OMX mge 5Y+	(87)	BMFN05Y	Finland	DS gov't 5Y
(40)	OMRXMAL	Sweden	OMX all mge	(88)	BMFN10Y	Finland	DS gov't 10Y
(41)	OMRXT30	Sweden	OMX t-bill 30d	(89)	SPBDEFI	Finland	S&P all gov't bonds
(42)	OMRXT60	Sweden	OMX t-bill 60d	(90)	SPFIZA	Finland	S&P gov't bonds 1Y+
(43)	RXMMSEK	Sweden	OMX MM index	(91)	TFNGVAL	Finland	DS all gov't tracker
(44)	SDOMBNC	Sweden	OMX t-bill benchmark	(92)	ADKGVAL	Denmark	DS gov't all mats
(45)	SDOMMRT	Sweden	OMX mge benchmark	(93)	ADKGVG1	Denmark	DS all gov't 1-3Y
(46)	SDOMOVN	Sweden	OMX overnight	(94)	ADKGVG2	Denmark	DS all gov't 3-5Y
(47)	SDOMRIR	Sweden	OMX real interest	(95)	ADKGVG3	Denmark	DS all gov't 5-7Y
(48)	SDOMRTB	Sweden	OMX mge and corp	(96)	ADKGVG4	Denmark	DS all gov't 7-10Y

Notes: This table details the Nordic bond indices used in the study. All data is retrieved from Refinitiv Eikon and the Eikon Code is given for each time series Abbreviations used; DS: Datastream, S&P: Standard & Poors, OMD: OMX Nordic Stock Exchange, ND: Nordea, SHB: Svenska Handelsbanken, Gov't: government, mge: mortgage, MM: money market, corp: corporate bonds, IG: investment grade, coll: collateralized mats: maturities, D: duration, Y: years to maturity, M: months to maturity, d: days to maturity.

Table A10: (Continued) Part 2
Detailed Overview of Bond-Indices

	Code	Country	Description		Code	Country	Description
(97)	ADKGVG5	Denmark	DS all gov't 10Y+	(145)	SPDKC15	Denmark	S&P coll 1-5Y
(98)	BMDK02Y	Denmark	DS gov't 2Y	(146)	SPDKC1D	Denmark	S&P coll 1Y+
(99)	BMDK03Y	Denmark	DS gov't 3Y	(147)	SPDKC2A	Denmark	S&P AA coll
(100)	BMDK05Y	Denmark	DS gov't 5Y	(148)	SPDKC35	Denmark	S&P coll 3-5Y
(101)	BMDK07Y	Denmark	DS gov't 7Y	(149)	SPDKC3A	Denmark	S&P AAA coll
(102)	BMDK10Y	Denmark	DS gov't 10Y	(150)	SPDKC51	Denmark	S&P coll 5-10Y
(103)	BMDK30Y	Denmark	DS gov't 30Y	(151)	SPDKC57	Denmark	S&P coll 5-7Y
(104)	NDGVT2Y	Denmark	ND gov't constant 2Y	(152)	SPDKC71	Denmark	S&P coll 7-10Y
(105)	NDGVT3Y	Denmark	ND gov't constant 3Y	(153)	SPDKGCD	Denmark	S&P all coll
(106)	NDGVT5Y	Denmark	ND gov't constant 5Y	(154)	SPDKGQD	Denmark	S&P quasi coll gov't
(107)	NDGVT7Y	Denmark	ND gov't constant 7Y	(155)	SPDKI1D	Denmark	S&P IG corp 1Y+
(108)	NDGVTAL	Denmark	ND all gov't	(156)	SPDKICD	Denmark	S&P all IG corp
(109)	NDMGC3Y	Denmark	ND callable mge 3Y	(157)	SPDKIDD	Denmark	S&P IG corp 7-10Y
(110)	NDMGC5Y	Denmark	ND callable mge 5Y	(158)	SPFIDKS	Denmark	S&P sovereign bonds
(111)	NDMGC7Y	Denmark	ND callable mge 7Y	(159)	SPSFIDK	Denmark	S&P sovereign bonds II
(112)	NDMGCAL	Denmark	ND all callable mge	(160)	TDKGVAL	Denmark	DS all mats tracker
(113)	NDMGCM2	Denmark	ND mge 2Y	(161)	TDKGVG1	Denmark	DS gov't 1-3Y tracker
(114)	NDMGCM3	Denmark	ND mge 3Y	(162)	TDKGVG2	Denmark	DS gov't 3-5Y tracker
(115)	NDMGCM5	Denmark	ND mge 5Y	(163)	TDKGVG3	Denmark	DS gov't 5-7Y tracker
(116)	NDMGCOM	Denmark	ND mge combined	(164)	TDKGVG4	Denmark	DS gov't 7-10Y tracker
(117)	NDMGEAL	Denmark	ND all mge	(165)	TDKGVG5	Denmark	DS gov't 10Y+ tracker
(118)	NDMGM3Y	Denmark	ND comb 3Y				
(119)	NDMGM5Y	Denmark	ND comb 5Y				
(120)	NDMGM7Y	Denmark	ND comb 7Y				
(121)	NYKMTAL	Denmark	NY all bond				
(122)	NYKMTGE	Denmark	NY all mge				
(123)	NYKMTLM	Denmark	NY long- index				
(124)	NYKMTLP	Denmark	NY long+ index				
(125)	NYKMTSH	Denmark	NY short index				
(126)	SPD3AGQ	Denmark	S&P AAA quasi gov't				
(127)	SPDGQ1D	Denmark	S&P quasi gov't 1Y+				
(128)	SPDI50D	Denmark	S&P IG corp 5-10Y				
(129)	SPDKA01	Denmark	S&P all bonds 0-1Y				
(130)	SPDKA10	Denmark	S&P all bonds 10Y+				
(131)	SPDKA13	Denmark	S&P all bonds 1-3Y				
(132)	SPDKA15	Denmark	S&P all bonds 1-5Y				
(133)	SPDKA2A	Denmark	S&P AA all bonds				
(134)	SPDKA35	Denmark	S&P all bonds 3-5Y				
(135)	SPDKA3A	Denmark	S&P AAA all bonds				
(136)	SPDKA3B	Denmark	S&P BBB all bonds				
(137)	SPDKA51	Denmark	S&P all bonds 5-10Y				
(138)	SPDKA57	Denmark	S&P all bonds 5-7Y				
(139)	SPDKA71	Denmark	S&P all bonds 7-10Y				
(140)	SPDKAG1	Denmark	S&P all bonds 1Y+				
(141)	SPDKAGN	Denmark	S&P all bonds				
(142)	SPDKC01	Denmark	S&P coll 0-1Y				
(143)	SPDKC10	Denmark	S&P coll 10Y+				
(144)	SPDKC13	Denmark	S&P coll 1-3Y				

Notes: This table details the Nordic bond indices used in the study. All data is retrieved from Refinitiv Eikon and the Eikon Code is given for each time series. Abbreviations used; DS: Datastream, S&P: Standard & Poors, OMD: OMX Nordic Stock Exchange, ND: Nordea, SHB: Svenska Handelsbanken, Gov't: government, mge: mortgage, MM: money market, corp: corporate bonds, IG: investment grade, coll: collateralized mats: maturities, D: duration, Y: years to maturity, M: months to maturity, d: days to maturity.

Table A11
Detailed Overview of Macroeconomic Bond Variables

	Underlying Characteristic		Transformations			Underlying Characteristic		Transformations	
	Acronym	Name	Agg.	tCode		Acronym	Name	Agg.	tCode
	Frictions					**Realized Variance**			
(1)	AMIHUD	Unscaled Amihud	Median	[4]	(16)	Jump	Jump Variation	Median	[5]
(2)	AMOUT	Amount Outstanding	Median	[4]	(17)	LT_BV	Long-Term Bipower Var.	Median	[4]
(3)	CBeta	Short-Term Credit Beta	Median	[5]	(18)	LT_Range	Long-Term Realized Range	Median	[2]
(4)	CMktBeta	Long-Term Credit Beta	Median	[4]	(19)	ST_BV	Short-Term Bipower Var.	Median	[4]
(5)	LME	Market Capitalization	Median	[4]	(20)	ST_Range	Short-Term Realized Range	Median	[5]
(6)	LT_Bond_Vol	Long-Term Volatility	Median	[5]	(21)	TQ	Tripower Quarticity	Median	[4]
(7)	ST_Bond_Vol	Short-Term Volatility	Median	[5]					
						Returns			
	Higher Moments				(22)	Bond_LT_Rev	Long-Term Reversal	Median	[2]
(8)	ES10	10% Expected Shortfall	Median	[4]	(23)	Bond_r12_3	3M Momentum	Median	[5]
(9)	ES5	5% Expected Shortfall	Median	[4]	(24)	Bond_r12_6	6M Momentum	Median	[5]
(10)	LT_Kurt	Long-Term Kurtosis	Median	[4]	(25)	Bond_r12_9	9M Momentum	Median	[5]
(11)	LT_Skew	Long-Term Skewness	Median	[5]	(26)	Bond_ST_Rev	Short-Term Reversal	Median	[2]
(12)	ST_Kurt	Short-Term Kurtosis	Median	[4]					
(13)	ST_Skew	Short-Term Skewness	Median	[5]		**Value**			
(14)	VaR10	10% Value-at-Risk	Median	[4]	(27)	CSpread	Credit Spread	Median	[5]
(15)	VaR5	5% Value-at-Risk	Median	[5]	(28)	Life	Life-to-Maturity	Median	[4]
					(29)	MD	Duration	Median	[5]
					(30)	YTM	Yield-to-Maturity	Median	[5]

Notes: This table details the transformations used to construct the macroeconomic bond characteristics. All underlying macroeconomic variables at time t are taken as cross-sectional medians of the non-imputed bond characteristics at any time t. For a detailed overview of the data sources and the construction of the underlying bond characteristics, see Table A10. Transformation codes (tCode) are as follows: [1] x_t; [2] Δx_t; [3] $\Delta \log(x_t)$; [4] $\Delta^2 \log(x_t)$; [5] $\Delta^2 x_t$.

A6. Result of Validation Procedure

Table A12
Machine Learning Model Hyperparameter Configurations

	Model Dimension	Hyperparameter	Grid Candidates	Selection
Generative Adversarial Network (GAN)				
(1)	Depth & Width	PK layer architecture	{[64, 64], [16, 16, 16]}	[16, 16, 16]
(2)	Width	Hidden states (k) in PK LSTM	$\{2, 4\}$	2
(3)	Width	Hidden states in COND LSTM	$\{2\}$	2
(4)	Non-linearity	PK FFN activation function	{ReLU, SELU}	SELU
(4)	Non-linearity	COND FFN activation function	{Linear}	Linear
(5)	Depth	Hidden layers in COND network	$\{0\}$	0
(6)	Depth	LSTM lookback weeks	$\{35\}$	35
(7)	Optimizer	Adam learning rate	$\{10^{-4}\}$	10^{-4}
(8)	Optimizer	Adam 1st moment decay (β_1)	$\{0.9\}$	0.9
(9)	Optimizer	Adam 2nd moment decay (β_2)	$\{0.999\}$	0.999
(10)	Regularization	Dropout Rate	$\{5\%, 10\%\}$	5%
(11)	Initialization	LSTM and Neural net initializer	Glorot (2010)	Glorot (2010)
(12)	Ensembling	Number of ensembles	$\{5\}$	5
Feedforward Network (FFN)				
(1)	Depth & Width	Layer architecture	$\{[32, 16, 8]$, [16, 16, 16]$\}$	[32, 16, 8]
(2)	Regularization	Dropout Rate	{5%, 20%}	20%
(3)	Non-linearity	Activation function	{ReLU, SeLU}	SeLU
(4)	Non-linearity	Include biases?	{Yes, No}	Yes
(5)	Ensembling	Number of ensembles	$\{5\}$	5
(6)	Optimizer	Adam learning rate	$\{10^{-3}\}$	10^{-3}
(7)	Optimizer	Adam 1st moment decay (β_1)	$\{0.9\}$	0.9
(8)	Optimizer	Adam 2nd moment decay (β_2)	$\{0.999\}$	0.999
(9)	Initialization	Neural net initializer	Glorot (2010)	Glorot (2010)
Elastic Net (EN)				
(1)	Regularization	Total penalty (α)	$[10^{-8}, 10^{-6}]$	$4.1 \cdot 10^{-8}$
(2)	Regularization	L1-ratio (λ)	$[10^{-2}, 1]$	0.066
(3)	Ensembling	Number of ensembles	$\{5\}$	5

Notes. This table illustrates the various hyperparameters of the generative adversarial network (GAN), the feedforward network (FFN) and the elatic net (EN) models. Candidate hyperparameters are illustrated in column three and the result of the validation procedure is presented in column four. A total 16 GAN models are compared on the validation dataset for the unconditional fit of Algorithm (1). A total of 16 models are validated for the FFN model and a total of $50^2 = 2,500$ models for the EN. The (α, λ)-grid for the EN is equally-log-spaced in both dimensions. The number of candidates for the GAN and FFN models are lower due to the computational intensity of the models, which makes it take a much longer time to train.

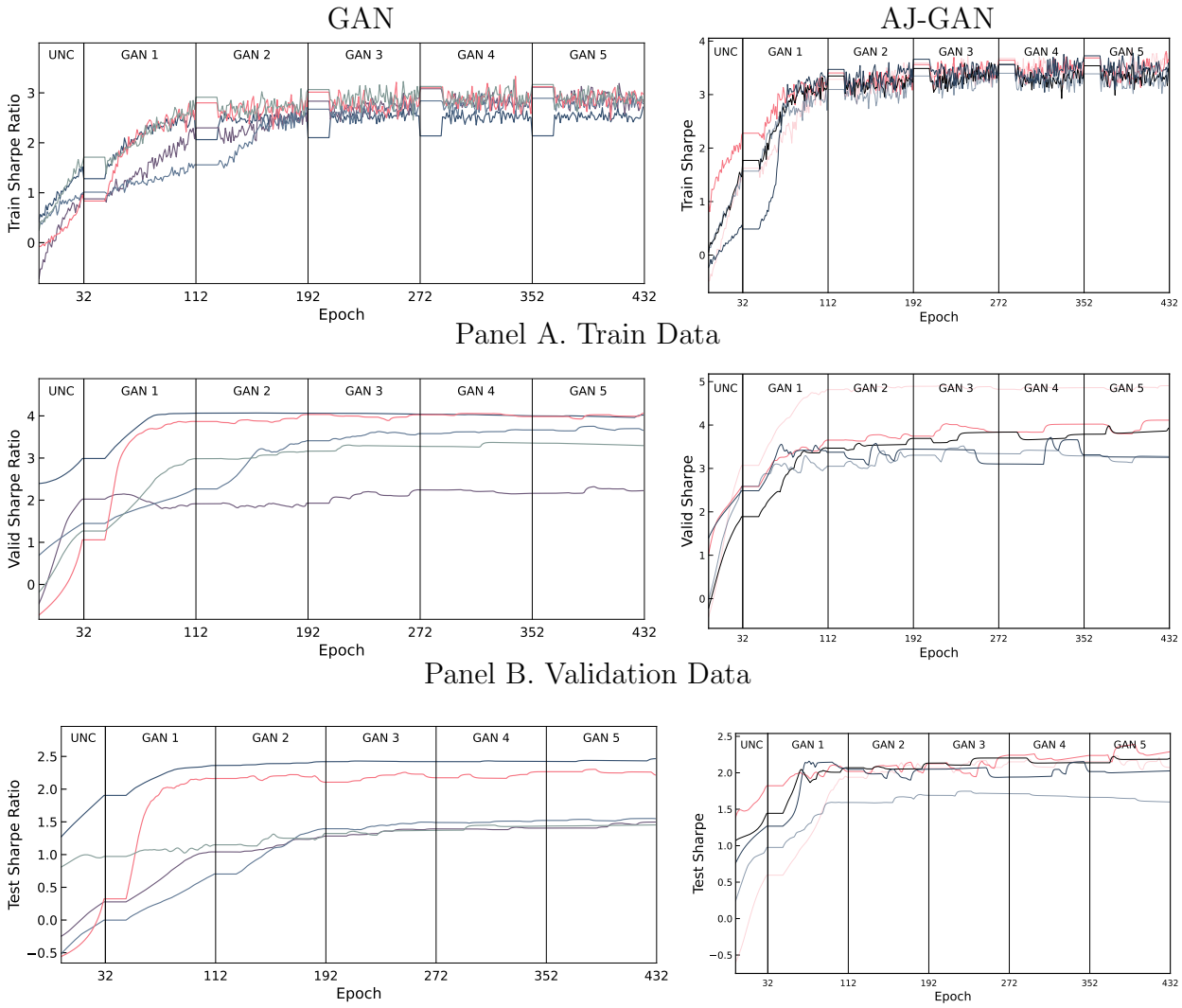


Figure A17

GAN and AJ-GAN Sharpe Ratio by Training Epoch

Notes. The figure illustrates the evolution of the Sharpe ratio for each training epoch of Algorithm (1) and Algorithm (2). The models are the baseline Chen et al. (2023a) generative adversarial network (GAN) and the Alvarez and Jermann (2005) version of the GAN model (AJ-GAN). The training epochs are separated into the two phases of the Algorithms, i.e., the unconditional fit (UNC) and then the conditional fits (GAN 1, GAN 2, etc.). Each solid line corresponds to one fit in the ensemble of the validated models, each initiated with a different seed. The horizontal lines at the start of each iteration correspond to the training of the conditioning network to find the test assets. The line is flat because the Sharpe ratio of the pricing kernel is defined only by the basis assets, and not by the test assets. For both the GAN and the AJ-GAN, the stable evolutionary equilibrium on the validation set is achieved by epoch 2 for most components. Because the out-of-sample performance never deteriorates, the final ensemble corresponds to the final epoch, i.e. the model configurations at epoch 432.

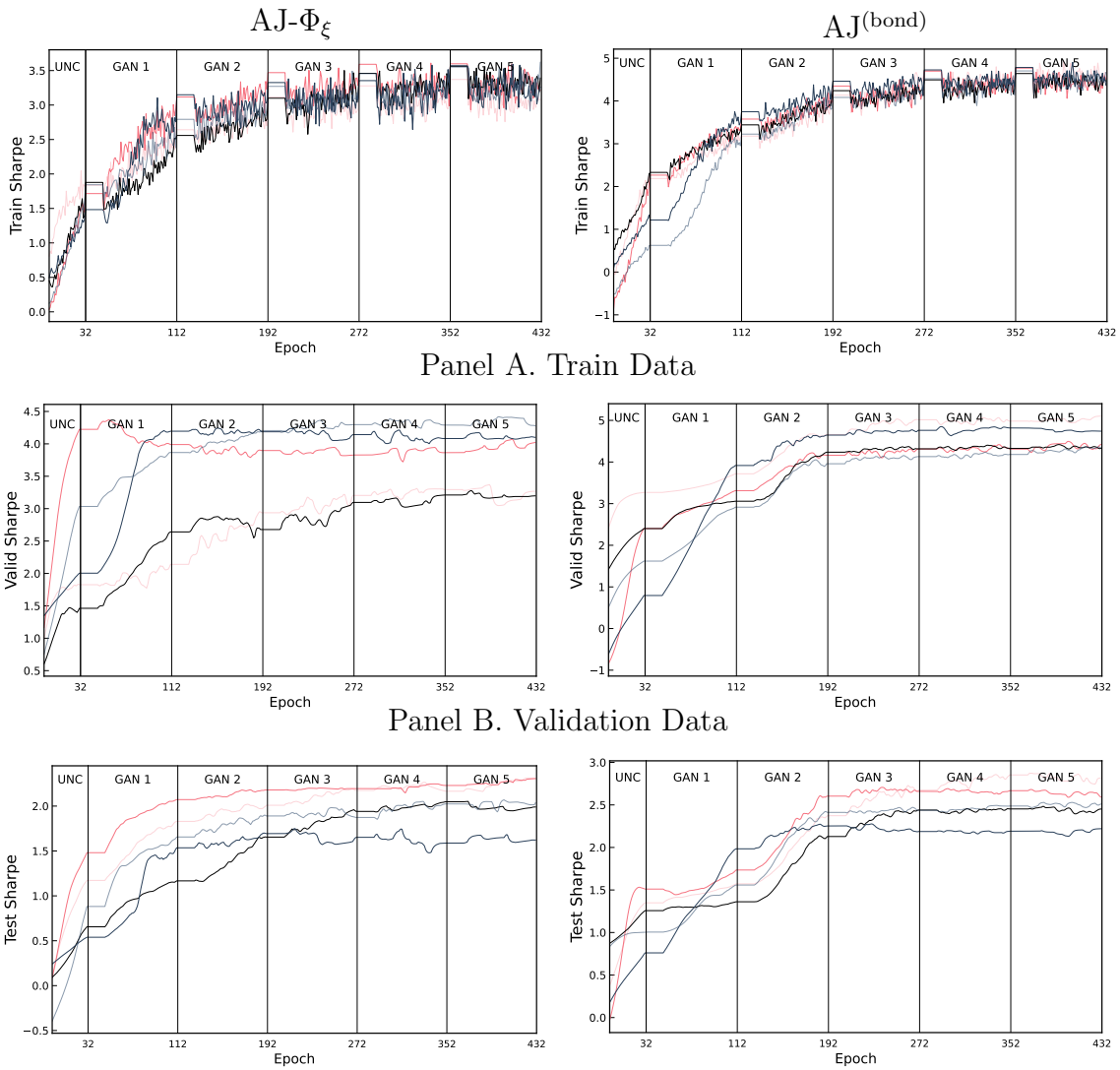


Figure A18

AJ-Φ_ξ and AJ$^{(\text{bond})}$ Sharpe Ratio by Training Epoch

Notes. The figure illustrates the evolution of the Sharpe ratio for each training epoch of Algorithm (2). The models are the Alvarez and Jermann (2005) version of the generative adversarial network with martingale constraint (AJ-Φ_ξ) and the one in which the transitory component is built using only bond data (AJ$^{(\text{bond})}$). The training epochs are separated into the two phases of Algorithm (2), i.e., the unconditional fit (UNC) and then the conditional fits (GAN 1, GAN 2, etc.). Each solid line corresponds to one fit in the ensemble of the validated models, each initiated with a different seed. The horizontal lines at the start of each iteration correspond to the training of the conditioning network to find the test assets. The line is flat because the Sharpe ratio of the pricing kernel is defined only by the basis assets, and not by the test assets. For the AJ-Φ_ξ model, the stable evolutionary equilibrium on the validation set is achieved by epoch 2 for most components. The convergence is in iteration 3 for the AJ$^{(\text{bond})}$ model. Because the out-of-sample performance never deteriorates, the final ensemble corresponds to the final epoch, i.e. the model configurations at epoch 432.

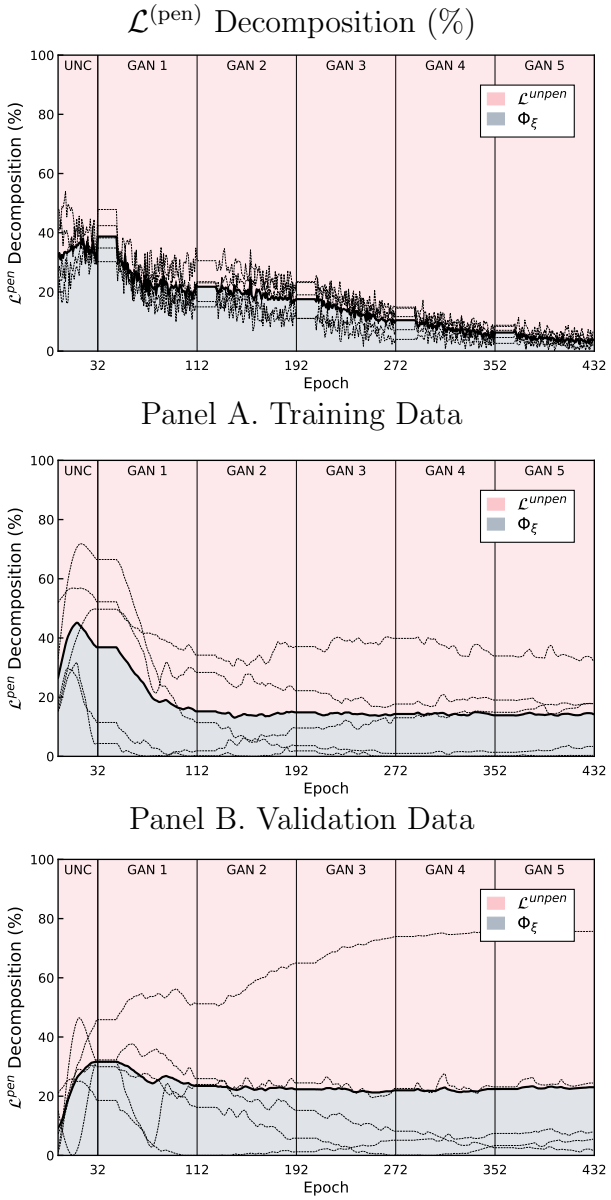


Figure A19

Decomposition of the Penalized Objective Function $\mathcal{L}^{(\text{pen})}$ in the AJ-Φ_ξ Model

Notes. This figure illustrates the percentage decomposition of the penalized loss $\mathcal{L}^{(\text{pen})}$ by training epoch for the Alvarez and Jermann (2005) version of the generative adversarial network with martingale regularization (AJ-Φ_ξ). The total penalized loss $\mathcal{L}^{(\text{pen})}$ is the sum of the unpenalized weighted sum of squared pricing errors $\mathcal{L}^{(\text{unpen})}$ (red shade) and the martingale regularization penalty $\Phi_\xi(M_{t+1}^{(\text{P})})$ with scaling parameter ξ (grey shade). The training epochs are separated into the two phases of Algorithm (2), i.e., the unconditional fit (UNC) and then the conditional fits (GAN 1, GAN 2, etc.). Dashed lines correspond to individual model fits within the final ensemble. The solid line correspond to the average of the individual model fits.

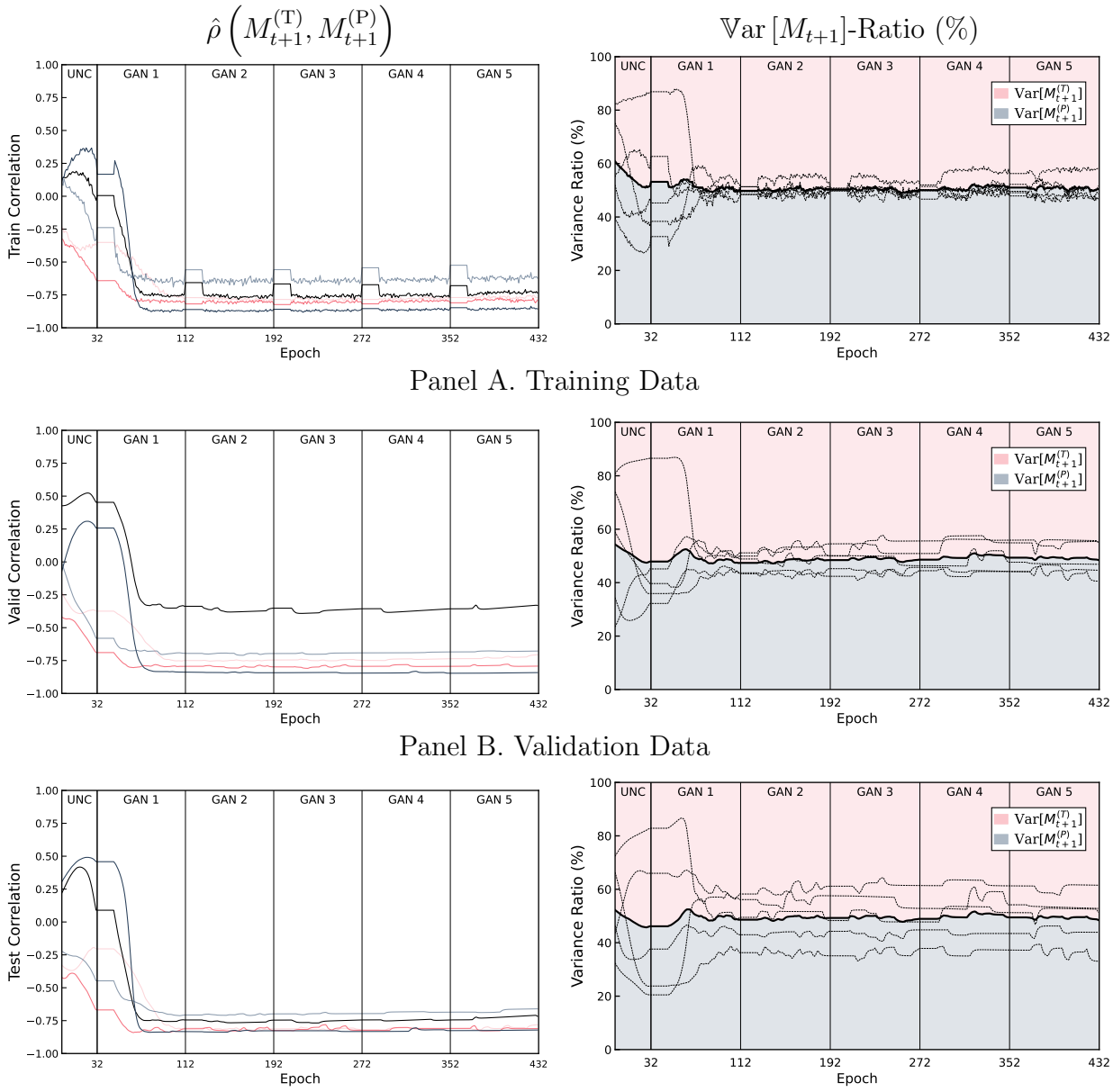


Figure A20

Correlation and Variance Ratio of AJ Component Kernels

Notes. The figure illustrates the empirical correlation of the permanent ($M_{t+1}^{(\mathrm{P})}$) and transitory ($M_{t+1}^{(\mathrm{T})}$) pricing kernel components (Left), and the ratio of the pricing kernel variances (Right) by training epoch of Algorithm (2). The pricing kernel is the one generated by the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN). The training epochs are separated into the two phases of Algorithm (2), i.e., the unconditional fit (UNC) and then the conditional fits (GAN 1, GAN 2, etc.). The variance ratio is the empirical version of the unconditional ratio $\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{P})}]/(\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{P})}] + \mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{T})}])$. Note that this ratio abstracts from other terms that part of the combined variance $\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{T})} \cdot M_{t+1}^{(\mathrm{P})}]$.

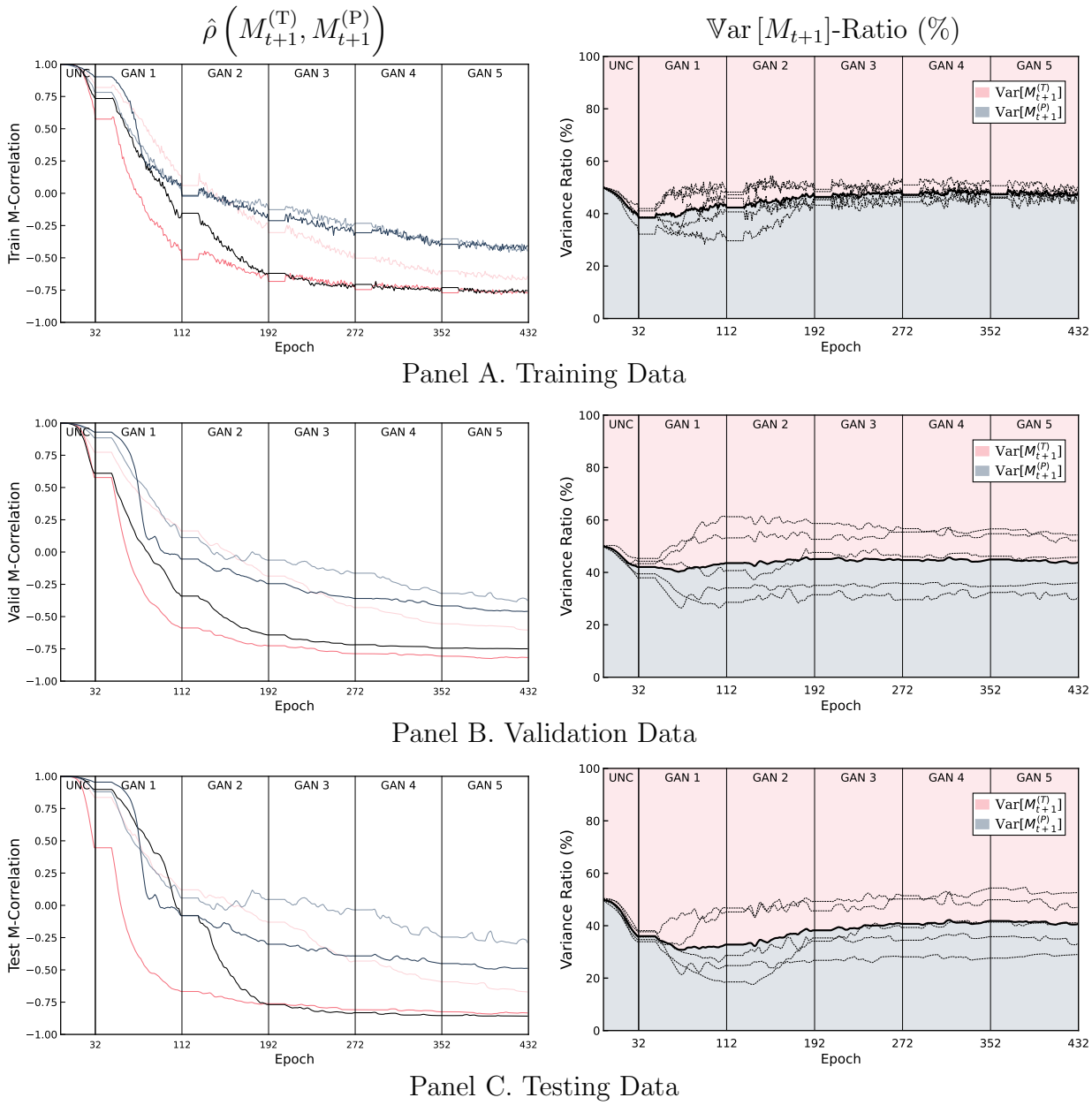


Figure A21

Correlation and Variance Ratio of AJ-Φ_ξ Component Kernels

Notes. The figure illustrates the empirical correlation of the permanent ($M_{t+1}^{(\mathrm{P})}$) and transitory ($M_{t+1}^{(\mathrm{T})}$) pricing kernel components (Left), and the ratio of the pricing kernel variances (Right) by training epoch of Algorithm (2). The pricing kernel is the one generated by the Alvarez and Jermann (2005) version of the generative adversarial network (GAN) that includes the martingale constraint (AJ-Φ_ξ). The training epochs are separated into the two phases of Algorithm (2), i.e., the unconditional fit (UNC) and then the conditional fits (GAN 1, GAN 2, etc.). The variance ratio is the empirical version of the unconditional ratio $\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{P})}]/(\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{P})}] + \mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{T})}])$. Note that this ratio abstracts from other terms that part of the combined variance $\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{T})} \cdot M_{t+1}^{(\mathrm{P})}]$. The perfect correlation at epoch 1 occurs because AJ-Φ_ξ is initialized using the same seeds for its two FFN networks.

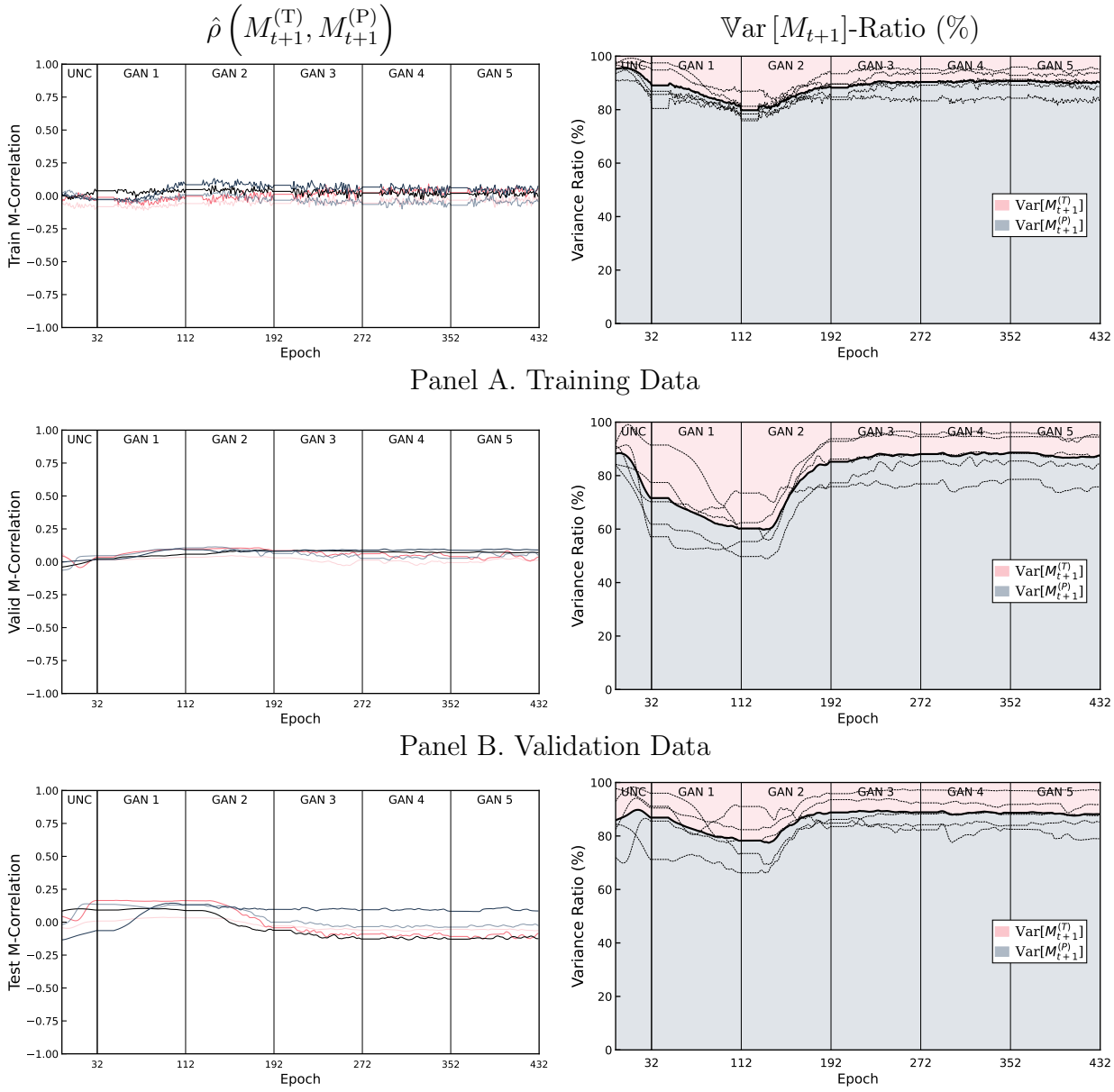


Figure A22

Correlation and Variance Ratio of $\mathrm{AJ}^{(\mathrm{bond})}$ Component Kernels

Notes. The figure illustrates the empirical correlation of the permanent ($M_{t+1}^{(\mathrm{P})}$) and transitory ($M_{t+1}^{(\mathrm{T})}$) pricing kernel components (Left), and the ratio of the pricing kernel variances (Right) by training epoch of Algorithm (2). The pricing kernel is the one generated by the Alvarez and Jermann (2005) version of the generative adversarial network (GAN) in which the transitory component is built using only bond data ($\mathrm{AJ}^{(\mathrm{bond})}$). The training epochs are separated into the two phases of Algorithm (2), i.e., the unconditional fit (UNC) and then the conditional fits (GAN 1, GAN 2, etc.). The variance ratio is the empirical version of the unconditional ratio $\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{P})}]/(\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{P})}]+\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{T})}])$. Note that this ratio abstracts from other terms that part of the combined variance $\mathbb{V}\mathrm{ar}[M_{t+1}^{(\mathrm{T})} \cdot M_{t+1}^{(\mathrm{P})}]$.

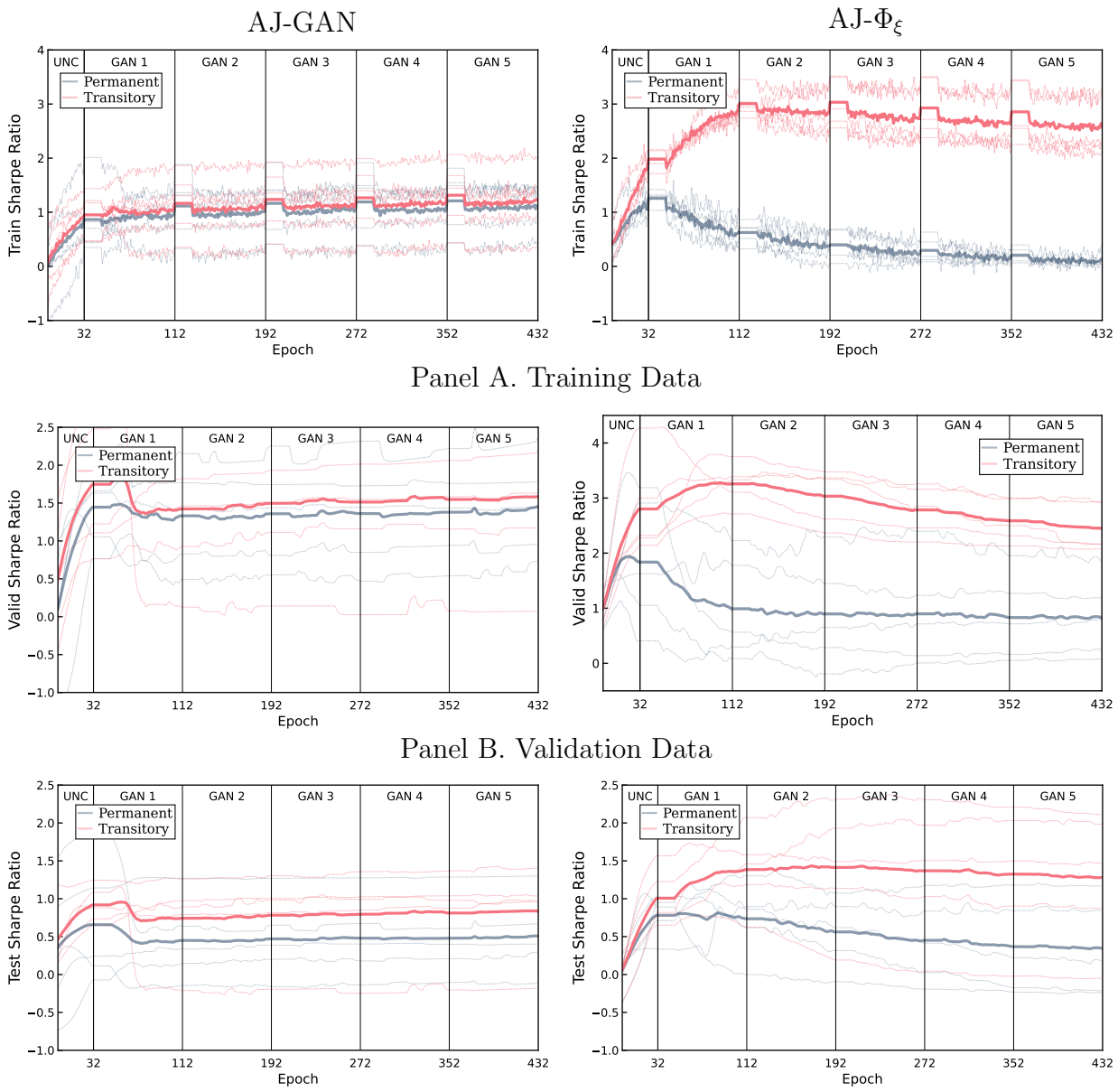
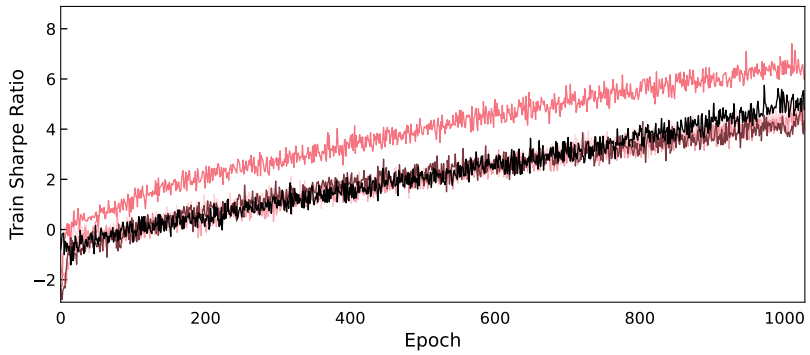


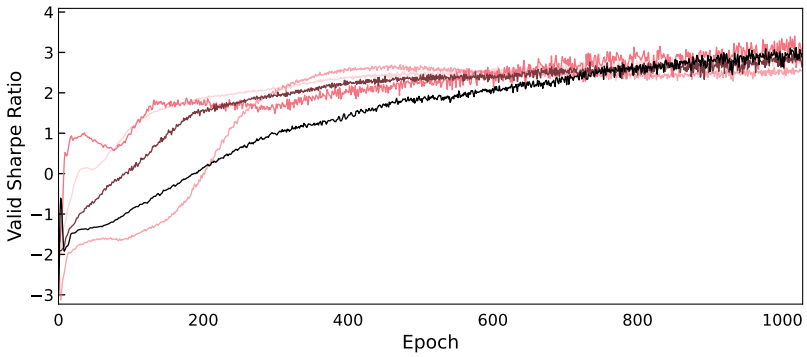
Figure A23

Sharpe Ratio of Permanent and Transitory Pricing Kernels

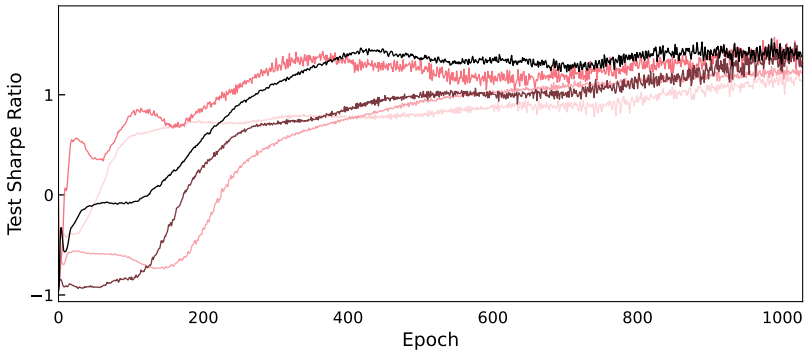
Notes. The figure illustrates the Sharpe ratios by training epoch of the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN) model (Left) and AJ-GAN with martingale constraint (AJ-Φ_ξ) (Right). The red line corresponds to the implied Sharpe ratio of the transitory pricing kernel $M_{t+1}^{(\mathrm{T})}$ and the blue line corresponds to the implied Sharpe ratio of the permanent pricing kernel $M_{t+1}^{(\mathrm{P})}$. Dashed lines correspond to ensemble component fits and solid lines correspond to average across ensemble.



(a) Train Data



(b) Validation Data



(c) Test Data

Figure A24
FFN Ensemble Sharpe Ratio by Training Epoch

Notes. The Figure illustrates the evolution of the Sharpe ratio for each epoch of the feedforward network (FFN) model of Gu et al. (2020). Each line correspond to one fit in the ensemble of validated models, each initiated with a different seed. Because the overall trend on the validation set does not imply severe overfitting, the final selected fits ahead of the ensembling procedure are the final fits, i.e. the model parameters at the final epoch 1028.

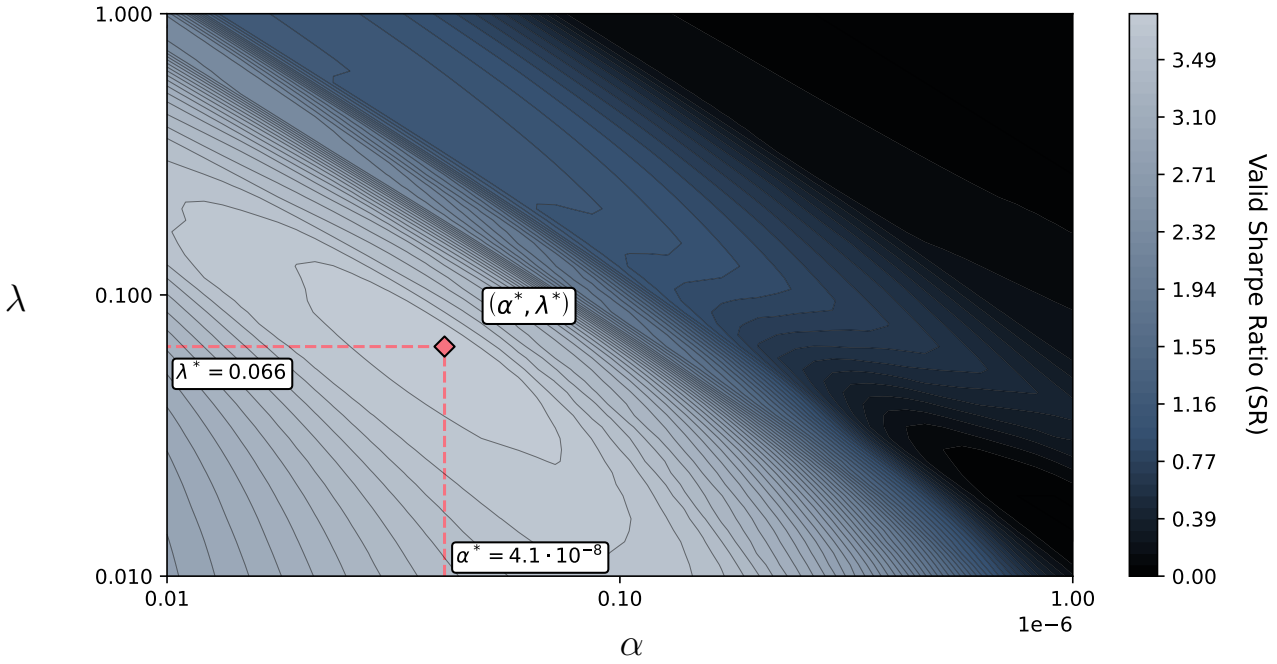


Figure A25

Elastic Net Validation Sharpe Ratio by α and λ

Notes. The figure illustrates level curves of the Sharpe Ratio (SR) of the elastic net model on the validation set during the hyperparameter selection of the overall penalty (α) and the Lasso penalty ratio (λ). The optimal model parameters $(\alpha^*, \lambda^*) = (4.1 \cdot 10^{-8}, 0.066)$ are extracted ahead of the ensembling procedure as this interior solution is associated with the highest Sharpe ratio. Sharpe ratios are downwards winsorized at zero. During the validation procedure the LARS is implemented with cyclical feature updates. For the ensemble implementation of LARS, a stochastic selection of feature updates is used in order to get different results for the same model configuration.

Table A13
Elastic Net Ensemble Fits

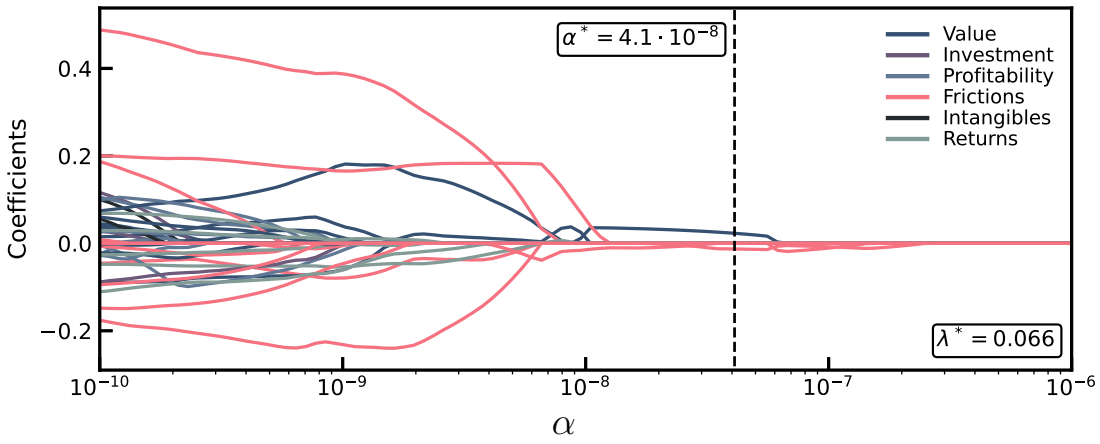
				Sharpe Ratio		
	Model	α	λ	Train	Valid	Test
(1)	Elastic Net 1	0.066	$4.1 \cdot 10^{-8}$	3.15	3.22	1.77
(2)	Elastic Net 2	0.066	$4.1 \cdot 10^{-8}$	3.29	3.53	1.95
(3)	Elastic Net 3	0.066	$4.1 \cdot 10^{-8}$	3.22	3.31	1.92
(4)	Elastic Net 4	0.066	$4.1 \cdot 10^{-8}$	3.21	3.32	1.82
(5)	Elastic Net 5	0.066	$4.1 \cdot 10^{-8}$	3.16	2.98	1.92
	Ensemble	0.066	$4.1 \cdot 10^{-8}$	3.23	3.31	1.90

Notes. This table illustrates the results of the individual fits within the elastic net ensemble. The ensemble model aggregates the portfolio weights of each elastic net fit as detailed in Section III.B. The corresponding results for the GAN and FFN are illustrated in Figures A17 and A24, respectively. The optimal pair (α, λ) is taken from the validation procedure presented in Figure A25.

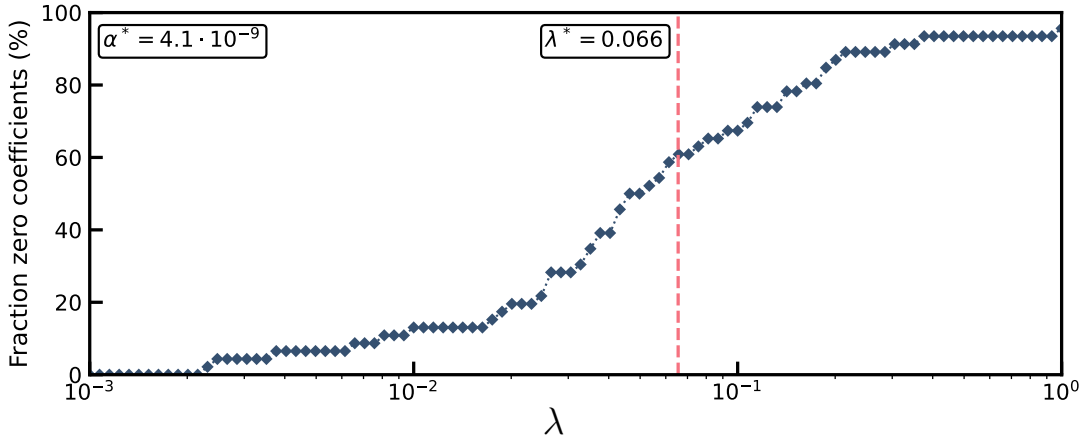
Table A14
GAN-Beta Network Hyperparameter Configuration

	Model Dimension	Hyperparameter	Grid Candidates	Selection
GAN-Beta Network $\left(\beta_{t,i}^{GAN}\right)$				
(1)	Depth & Width	Layer architectures	{[64, 64], [16, 16, 16]}	[64, 64]
(2)	Regularization	Dropout rate	{10%, 35%}	35%
(3)	Non-linearity	Activation function	{Linear, SELU}	SELU
(4)	Optimizer	Adam learning rate	$\{10^{-4}\}$	10^{-4}
(5)	Optimizer	Adam 1st moment decay (β_1)	{0.9}	0.9
(6)	Optimizer	Adam 2nd moment decay (β_2)	{0.999}	0.999

Notes. This table illustrates the various hyperparameters of the beta network for the generative adversarial network (GAN) model. Candidate hyperparameters are illustrated in column three and the result of the validation procedure is presented in column four. A total $2^3 = 8$ GAN-Beta network models are compared on the validation dataset after 128 epochs. The best model on the validation data is then re-estimated five times for different seeds and ensembled into an aggregated final estimate of the function $\beta_{t,i}^{GAN} : \mathcal{I}_t \to \mathbb{R}$



Panel A. Elastic Net Coefficients by Penalty α



Panel B. Number of Non-Zero EN Coefficients by Mixing Parameter λ

Figure A26
Shrinkage in the Elastic Net Model

Notes. Panel A illustrates the estimated shrinkage on coefficients for the Elastic Net model holding constant the optimal mixing parameter λ as found in Figure A25 and Table A12, but varying the overall penalization α. Shrinkage affects all characteristic categories but frictions are the least affected, implying these characteristics are the core of the economic content extracted by the model. Panel B illustrates the number of coefficients that are regularized away from the model by holding constant the optimal α and re-estimating the EN for varying λ.

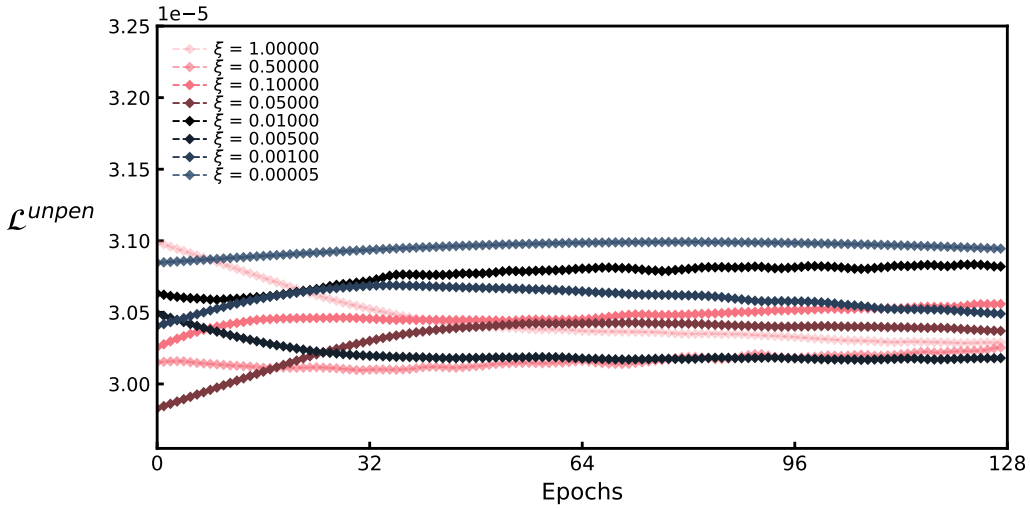


Figure A27

Validation Results for the ξ-Parameter in the AJ-Φ_ξ Model.

Notes. This figure illustrates the results of the validation procedure for the scaling parameter ξ in the Alvarez and Jermann (2005) version of the generative adversarial network with martingale constraint AJ-Φ_ξ. The y-axis plots the unpenalized objective function $\mathcal{L}^{(\text{unpen})}$ for the validation set and the x-axis plots the 128 training epochs of the unconditional fit in Algorithm (2). The scaling value $\xi = 0.005$ ends up with the lowest unpenalized loss at the end of the training period. This value is chosen going into the empirical estimation of the AJ-Φ_ξ model.

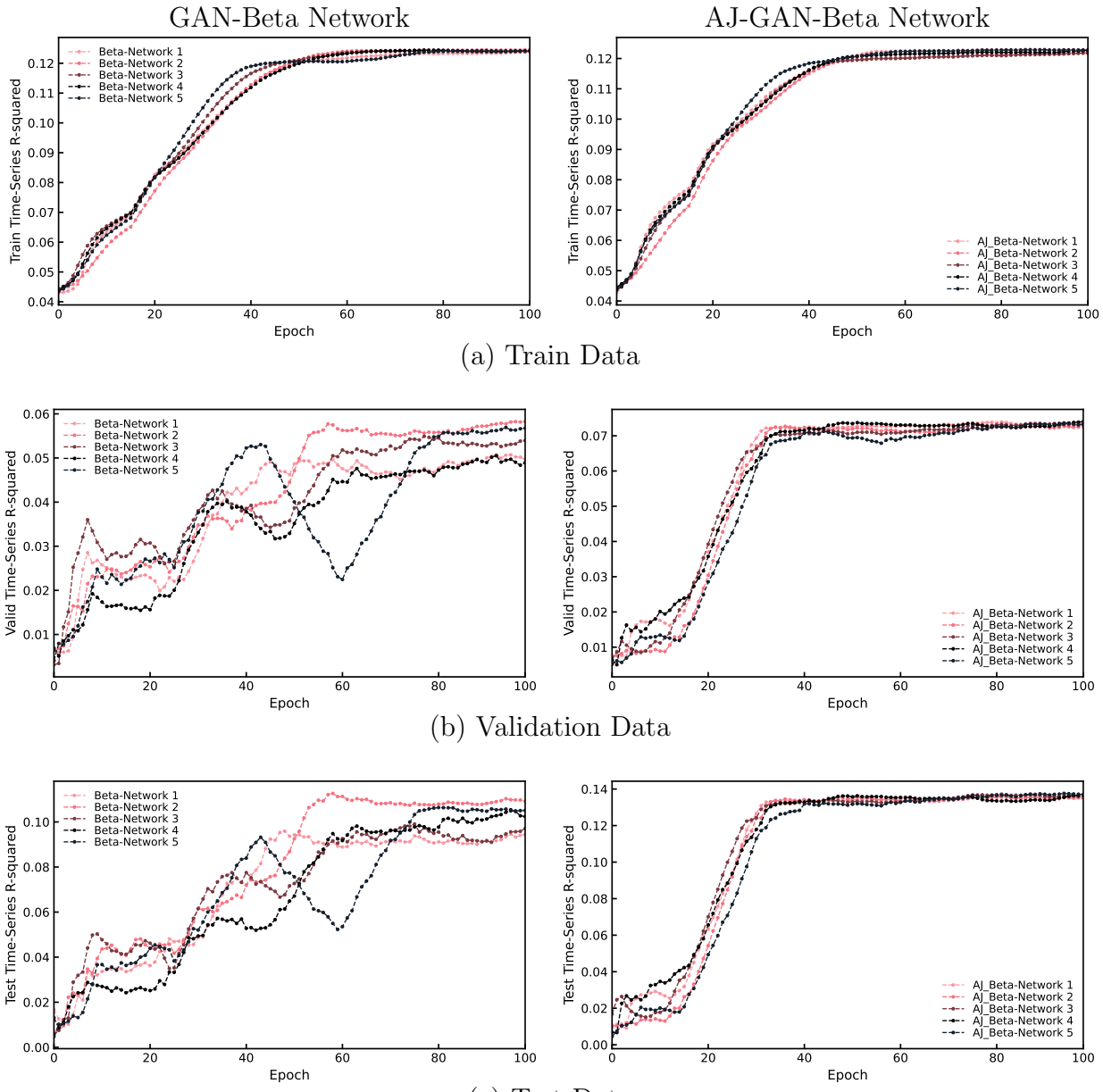
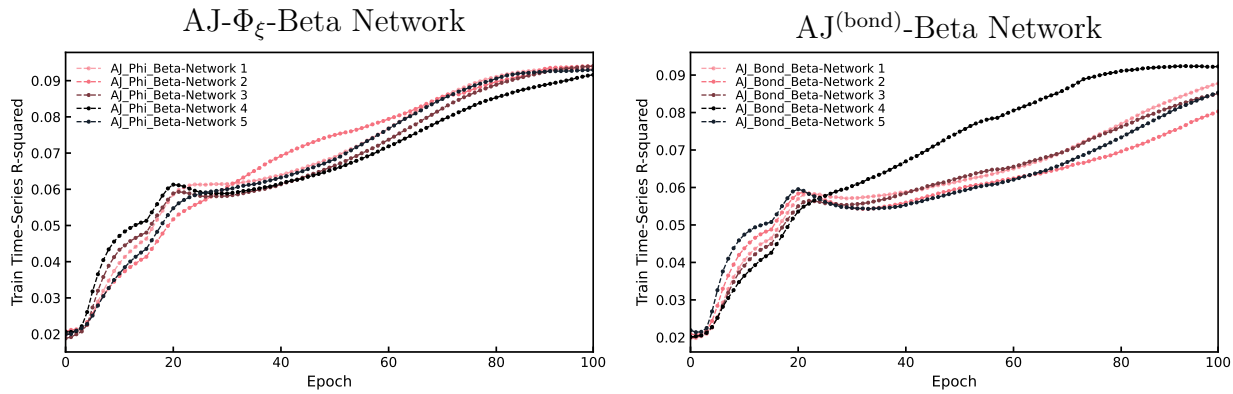


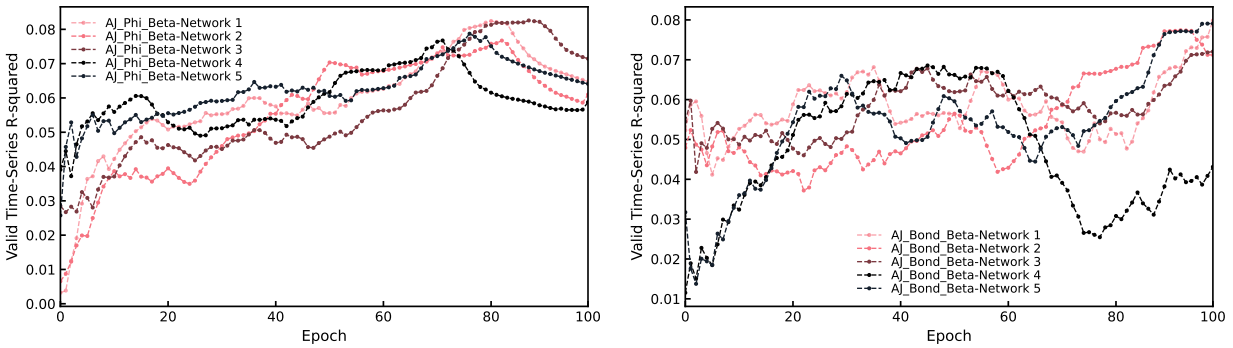
Figure A28

GAN-Beta and AJ-GAN-Beta TS-R^2 by Training Epoch

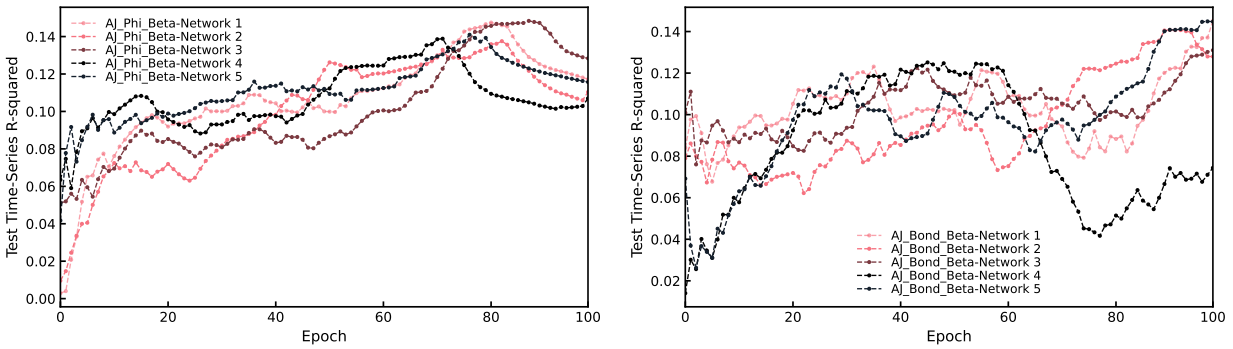
Notes. This figure illustrates the time series explained variation (TS-R^2) by training epoch for the beta neural to the network generative adversarial network (left) and the beta neural network for the Alvarez and Jermann (2005) version of the GAN model (right). The final ensemble fit is the average of the final model configurations at epoch 100. Note that the training set includes the effect of the dropout mask, which creates a wedge between the TS-R^2 on the training set during training and when evaluated.



(a) Train Data



(b) Validation Data



(c) Test Data

Figure A29

AJ-Φ_ξ-Beta and AJ$^{(\text{bond})}$-Beta TS-R^2 by Training Epoch

Notes. This figure illustrates the time series explained variation (TS-R^2) by training epoch for the beta neural network to the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN) with the martingale constraint (left) and the beta neural network for the AJ-GAN model in which the transitory component is built using only bond data (right). The final ensemble fit is the average of the final model configurations at epoch 100. Note that the training set includes the effect of the dropout mask, which creates a wedge between the TS-R^2 on the training set during training and when evaluated.

A7. Fama-French Robustness Checks

Table A15
GMM Results for the Fama-French Factor Models

	Fama French 3				Fama French 5			
	$\boldsymbol{\lambda}$	t	[2.5%	97.5%]	$\boldsymbol{\lambda}$	t	[2.5%	97.5%]
$f_t^{(\text{MKT})}$	1.002	142.7	0.988	1.016	1.011	130.8	0.995	1.026
$f_t^{(\text{SMB})}$	0.712	61.8	0.689	0.734	0.719	58.6	0.695	0.743
$f_t^{(\text{HML})}$	0.490	52.2	0.472	0.509	0.289	30.3	0.271	0.308
$f_t^{(\text{RMW})}$					0.135	14.0	0.116	0.154
$f_t^{(\text{CMA})}$					-0.082	-8.7	-0.100	-0.064
	Sharpe ratio (SR)				Sharpe ratio (SR)			
	Full	Train	Valid	Test	Full	Train	Valid	Test
	0.58	0.30	1.38	0.80	0.55	0.27	1.39	0.72

Notes. The table presents the generalized method of moments (GMM) loading estimates $\boldsymbol{\lambda}$ for the Fama-French 3-factor model (Fama and French, 1992) and the Fama-French 5-factor model (Fama and French, 2015). The t statistic and the 95% confidence intervals are also presented. The annualized Sharpe ratio (SR) of the empirical return factor $F_{t+1}^{(\text{FF})} = \boldsymbol{\lambda}'\boldsymbol{f}_t$ is presented for each sub-sample. Newey-West standard errors are used to correct for heteroskedasticity and autocorrelation (using $T^{1/4} \cdot 0.75 \approx 2$ lags).

This section provides robustness checks for the CAPM regressions and GRS tests in Table II by performing the same analysis with the Fama and French (1992) 3-factor model (FF3) and the Fama and French (2015) 5-factor models (FF5). The Fama-French model specifies the pricing kernel using the following factor structure:

$$M_{t+1}^{(\text{FF})} = 1 - \boldsymbol{\lambda}'\boldsymbol{f}_t \tag{30}$$

where $\boldsymbol{f}_t$ is a vector that contains excess returns on empirical risk factors and $\boldsymbol{\lambda}$ is a vector of loadings on the factors. Specifically, FF5 uses the market factor ($f_t^{(\text{MKT})}$), the size factor ($f_t^{(\text{SMB})}$) the value factor($f_t^{(\text{HML})}$), the profitability factor ($f_t^{(\text{RMW})}$), and investment factor ($f_t^{(\text{CMA})}$):

$$\boldsymbol{f}_t^{(\text{FF5})} = \begin{bmatrix} f_t^{(\text{MKT})} \\ f_t^{(\text{SMB})} \\ f_t^{(\text{HML})} \\ f_t^{(\text{RMW})} \\ f_t^{(\text{CMA})} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\lambda}^{(\text{FF5})} = \begin{bmatrix} \gamma_t^{(\text{MKT})} \\ \gamma_t^{(\text{SMB})} \\ \gamma_t^{(\text{HML})} \\ \gamma_t^{(\text{RMW})} \\ \gamma_t^{(\text{CML})} \end{bmatrix} \tag{31}$$

The FF3 model includes only a subset of the FF5 factors, i.e. $f_t^{(\text{MKT})}$, $f_t^{(\text{SMB})}$, and $f_t^{(\text{HML})}$. Empiricially, the factor returns are constructed from variables available in the firm characteristics information set $I_{t,i}$, see Table A20. The loading parameters $\boldsymbol{\lambda}$ are estimated using the basic generalized method of moments (GMM) procedure of Hansen (1982):

$$\bar{g}\left(\boldsymbol{\lambda}\right) = \frac{1}{T}\sum_{t=1}^{T} g_t(\boldsymbol{\lambda}) = \mathbf{0}_{q\times 1} \tag{32}$$

where $\bar{g}(\cdot)$ is the sample average function. The number of moment equations k and parameters q is the same, i.e., $q = k = 3$ in the FF3 model and $q = k = 5$ in the FF5 model. This means that the GMM estimation is exactly identified and that no explicit minimization of the GMM estimator $\hat{\boldsymbol{\lambda}} \in \arg\min \bar{g}(\boldsymbol{\lambda})' W \bar{g}(\boldsymbol{\lambda})$ is required because all sample conditions can be set directly to zero $\bar{g}(\hat{\boldsymbol{\lambda}}) = \mathbf{0}_{q\times 1}$ (Singleton, 2005). Here, W is a positive symmetric definite weighting matrix of dimension $q \times q$. Although not necessary, I apply a numerical optimization algorithm to solve the equations of k moments g_t implied by the orthogonality constraint:[43]

$$g_t = \boldsymbol{f}_t\left(1 - \boldsymbol{\lambda}'\boldsymbol{f}_t\right) R_{t+1}^{(\text{test})} \tag{33}$$

Table A15 presents the vector estimate $\boldsymbol{\lambda}$ and the empirical Sharpe ratios for the FF3 and FF5 models. The estimate is created using only training data and these fixed loadings are used out of sample to produce the empirical Sharpe ratios of the return factor $F_{t+1}^{(\text{FF})} = \boldsymbol{\lambda}'\boldsymbol{f}_t$ on the validation and test sets. For each subsample the FF3 and FF5 pricing kernels have a larger Sharpe ratio than the standalone market factor. The market factor had a Sharpe ratio of 0.05 (train), 1.32 (valid), and 0.72 (test). Using the out-of-sample Sharpe ratio of the Fama-French factors $F_{t+1}^{(\text{FF})}$ and regressions of the decile-sorted portfolios on loadings β from all models, the Gibbons et al. (1989) asset pricing test can be performed for the Fama-French models. Table A16 and Table A17 present these results for the GAN, FFN, and EN models on the FF3 and FF5 benchmarks, respectively. The null of the GRS that the FF3 and FF5 model correctly prices the decile-sorted portfolios generated by the GAN and EN are still rejected at the 5% level. Table A18 and Table A19 presents the same FF3 and FF5 asset pricing tests for the AJ-GAN, AJ-Φ_ξ, and AJ$^{(\text{bond})}$ models. The null hypothesis can be rejected at the 5% confidence level for all 3 models. Therefore, the CAPM benchmark results presented in the main text are also robust to the commonly used FF3 and FF5 benchmarks.

[43] I adopt the derivative-free simplex method of Nelder and Mead (1965) to do the minimization.

Table A16
Pricing Performance on Decile Sorted β-Portfolios versus FF3

	Test Data (2016 - 2023)											
	GAN				FFN				EN			
Decile	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p
1	6.9	-10.8	-2.59	0.01***	8.0	-6.8	-2.80	0.01***	7.9	-10.5	-3.08	0.00***
2	17.0	-1.7	-0.59	0.56	10.5	-4.5	-2.00	0.05**	14.9	-3.7	-1.18	0.24
3	13.2	-5.6	-1.95	0.05*	15.5	0.3	0.17	0.87	14.1	-4.7	-1.51	0.13
4	14.9	-3.7	-1.21	0.23	13.4	-1.5	-0.73	0.47	17.3	-1.6	-0.60	0.55
5	20.7	1.7	0.64	0.53	12.7	-2.4	-1.11	0.27	13.6	-5.2	-1.86	0.06*
6	18.5	-0.5	-0.16	0.87	16.5	1.4	0.65	0.52	18.7	-0.1	-0.03	0.98
7	25.0	6.2	1.87	0.06*	16.4	1.2	0.53	0.59	19.0	0.3	0.11	0.91
8	24.8	6.3	1.69	0.09*	17.5	2.6	1.00	0.32	24.8	6.0	1.91	0.06*
9	28.2	9.8	2.34	0.02**	17.8	3.1	1.10	0.27	24.8	6.0	1.91	0.06*
10	29.7	12.0	2.66	0.01***	22.2	8.2	2.34	0.02**	32.2	13.6	4.22	0.00***
10-1	22.8	22.9	3.06	0.00***	14.2	15.0	3.46	0.00***	24.3	24.1	5.19	0.00***
GRS AP Test			χ^2_{10}	p			χ^2_{10}	p			χ^2_{10}	p
			21.1	0.021**			17.0	0.074*			35.3	0.000***

Notes. This table reports the results of the asset pricing tests versus the Fama and French (1992) 3-factor model for the generative adversarial network (GAN), the feedforward network (FFN), and the elastic net (EN). Each model generates risk loadings $\beta_{t,i}$ for all stocks. These loadings are sorted by deciles every week, and equally weighted portfolios are constructed within deciles. The sample estimate of the unconditional expected return for each portfolio is annualized and presented under the heading $\mathbb{E}[R^e_{t+1}]$. The annualized pricing error (α), its t-stat (t) and the two-tailed p-value (p) are calculated from a projection of portfolio returns onto the returns of the market factor ($f_t^{(\text{MKT})}$), size factor ($f_t^{(\text{SMB})}$), and the value factor ($f_t^{(\text{HML})}$). Excess returns and pricing errors are expressed as percentage units. The GRS asset pricing (AP) test statistic is defined in Gibbons et al. (1989) and has the null hypothesis that the FF3 model correctly prices all decile portfolios. Newey-West standard errors are used to correct for heteroskedasticity and autocorrelation (using $T^{1/4} \cdot 0.75 \approx 2$ lags). Significance levels are specified as follows: * $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$.

Table A17
Pricing Performance on Decile Sorted β-Portfolios versus FF5

	Test Data (2016 - 2023)											
	GAN				FFN				EN			
Decile	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p
1	6.9	-8.4	-2.13	0.03**	8.0	-5.9	-2.41	0.02**	7.9	-9.0	-2.73	0.01***
2	17.0	0.1	0.04	0.97	10.5	-4.0	-1.79	0.07*	14.9	-2.6	-0.83	0.41
3	13.2	-4.2	-1.51	0.13	15.5	0.9	0.45	0.65	14.1	-3.5	-1.10	0.27
4	14.9	-2.8	-0.92	0.36	13.4	-0.9	-0.46	0.65	17.3	-0.7	-0.26	0.79
5	20.7	2.7	1.03	0.30	12.7	-1.8	-0.83	0.41	13.6	-4.4	-1.58	0.12
6	18.5	0.3	0.11	0.91	16.5	2.0	0.93	0.35	18.7	0.7	0.25	0.80
7	25.0	6.2	1.93	0.05*	16.4	2.2	1.02	0.31	19.0	1.2	0.39	0.70
8	24.8	6.4	1.73	0.09*	17.5	3.0	1.16	0.25	24.8	7.0	2.29	0.02**
9	28.2	9.5	2.36	0.02**	17.8	4.1	1.46	0.14	24.8	6.9	2.19	0.03**
10	29.7	11.5	2.65	0.01***	22.2	9.3	2.64	0.01***	32.2	13.9	4.40	0.00***
10-1	22.8	19.9	2.85	0.00***	14.2	15.2	3.46	0.00***	24.3	23.0	5.00	0.00***
GRS AP Test			χ^2_{10}	p			χ^2_{10}	p			χ^2_{10}	p
			19.9	0.031**			17.7	0.061*			34.6	0.000***

Notes. This table reports the results of the asset pricing tests versus the Fama and French (2015) 5-factor model for the generative adversarial network (GAN), the feedforward network (FFN), and the elastic net (EN). Each model generates risk loadings $\beta_{t,i}$ for all stocks. These loadings are sorted by deciles every week, and equally weighted portfolios are constructed within deciles. The sample estimate of the unconditional expected return for each portfolio is annualized and presented under the heading $\mathbb{E}[R^e_{t+1}]$. The annualized pricing error (α), its t-stat (t) and the two-tailed p-value (p) are calculated from a projection of portfolio returns onto the returns of the market factor ($f_t^{(\text{MKT})}$), size factor ($f_t^{(\text{SMB})}$), the value factor ($f_t^{(\text{HML})}$), the profitability factor ($f_t^{(\text{RMW})}$), and the investment factor ($f_t^{(\text{CMA})}$). Excess returns and pricing errors are expressed as percentage units. The GRS asset pricing (AP) test statistic is defined in Gibbons et al. (1989) and has the null hypothesis that the FF5 model correctly prices all decile portfolios. Newey-West standard errors are used to correct for heteroskedasticity and autocorrelation (using $T^{1/4} \cdot 0.75 \approx 2$ lags). Significance levels are specified as follows: * $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$.

Table A18
Pricing Performance on Decile Sorted AJ-β-Portfolios versus FF3

	Test Data (2016 - 2023)											
	AJ-GAN				AJ-Φ_ξ				AJ$^{(\text{bond})}$			
Decile	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p
1	7.9	-3.2	-0.70	0.48	2.7	-6.2	-1.98	0.05**	5.6	-6.1	-1.72	0.09*
2	10.2	-7.6	-2.26	0.02**	6.5	-3.6	-1.61	0.11	9.8	-4.7	-1.67	0.10*
3	15.8	-6.2	-1.91	0.06*	9.2	-4.4	-2.04	0.04**	14.3	-3.9	-1.51	0.13
4	14.2	-4.5	-1.57	0.12	9.8	-7.8	-3.42	0.00***	18.0	-2.2	-0.80	0.43
5	19.0	4.1	1.50	0.13	15.8	-2.9	-1.18	0.24	22.0	-1.2	-0.44	0.66
6	20.8	0.2	0.06	0.95	19.1	-2.1	-0.77	0.44	20.1	-2.2	-0.68	0.50
7	23.0	2.5	0.91	0.36	23.5	6.4	2.59	0.01***	24.3	2.4	0.77	0.44
8	25.6	4.2	1.70	0.09*	27.9	5.1	1.74	0.08*	32.2	7.7	1.91	0.06*
9	26.9	7.0	2.40	0.02**	31.8	13.2	3.81	0.00***	29.2	8.0	2.07	0.04**
10	29.3	3.5	1.16	0.25	36.9	20.8	5.09	0.00***	37.3	14.0	3.17	0.00***
10-1	21.4	6.7	1.18	0.24	34.2	26.9	4.53	0.00***	31.7	20.1	3.40	0.00***
GRS AP Test			χ^2_{10}	p			χ^2_{10}	p			χ^2_{10}	p
			20.8	0.023**			57.1	0.000***			21.0	0.021**

Notes. This table reports the results of the asset pricing tests versus the Fama and French (1992) 3-factor model for the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN model with the martingale penalty term (AJ-Φ_ξ), and the AJ-GAN model in which the temporary component is built from a portfolio of bond indices (AJ$^{(\text{bond})}$). Each model generates risk loadings $\beta_{t,i}$ for all stocks. These loadings are sorted by deciles every week, and equally weighted portfolios are constructed within deciles. The sample estimate of the unconditional expected return for each portfolio is annualized and presented under the heading $\mathbb{E}[R^e_{t+1}]$. The annualized pricing error (α), its t-stat (t) and the two-tailed p-value (p) are calculated from a projection of portfolio returns onto the returns of the market factor ($f^{(\text{MKT})}_t$), size factor ($f^{(\text{SMB})}_t$), and the value factor ($f^{(\text{HML})}_t$).. Excess returns and pricing errors are expressed as percentage units. The GRS asset pricing (AP) test statistic is defined in Gibbons et al. (1989) and has the null hypothesis that the FF3 model correctly prices all decile portfolios. Newey-West standard errors are used to correct for heteroskedasticity and autocorrelation (using $T^{1/4} \cdot 0.75 \approx 2$ lags). Significance levels are specified as follows: * $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$.

Table A19
Pricing Performance on Decile Sorted AJ-β-Portfolios versus FF3

	Test Data (2016 - 2023)											
	AJ-GAN				AJ-Φ_ξ				AJ$^{(\text{bond})}$			
Decile	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p	$\mathbb{E}[R^e_{t+1}]$	α	t	p
1	7.9	-0.7	-0.16	0.88	2.7	-4.7	-1.55	0.12	5.6	-5.0	-1.42	0.16
2	10.2	-6.0	-1.74	0.08*	6.5	-2.3	-1.04	0.30	9.8	-3.5	-1.28	0.20
3	15.8	-4.8	-1.52	0.13	9.2	-3.5	-1.64	0.10	14.3	-2.9	-1.12	0.26
4	14.2	-3.4	-1.19	0.24	9.8	-7.0	-3.13	0.00***	18.0	-1.1	-0.39	0.69
5	19.0	5.0	1.88	0.06*	15.8	-2.1	-0.89	0.37	22.0	-0.5	-0.21	0.84
6	20.8	0.5	0.20	0.84	19.1	-1.7	-0.64	0.52	20.1	-1.5	-0.45	0.65
7	23.0	3.0	1.10	0.27	23.5	6.6	2.66	0.01***	24.3	3.3	1.09	0.28
8	25.6	4.6	1.91	0.06*	27.9	5.6	1.92	0.06*	32.2	7.8	1.98	0.05**
9	26.9	7.4	2.57	0.01**	31.8	13.4	3.98	0.00***	29.2	9.0	2.33	0.02**
10	29.3	3.8	1.28	0.20	36.9	20.4	5.18	0.00***	37.3	15.2	3.37	0.00***
10-1	21.4	4.5	0.80	0.43	34.2	25.1	4.46	0.00***	31.7	20.2	3.40	0.00***
GRS AP Test			χ^2_{10}	p			χ^2_{10}	p			χ^2_{10}	p
			19.5	0.034**			56.3	0.000***			21.5	0.018**

Notes. This table reports the results of the asset pricing tests versus the Fama and French (2015) 5-factor model for the Alvarez and Jermann (2005) version of the generative adversarial network (AJ-GAN), the AJ-GAN model with the martingale penalty term (AJ-Φ_ξ), and the AJ-GAN model in which the temporary component is built from a portfolio of bond indices (AJ$^{(\text{bond})}$). Each model generates risk loadings $\beta_{t,i}$ for all stocks. These loadings are sorted by deciles every week, and equally weighted portfolios are constructed within deciles. The sample estimate of the unconditional expected return for each portfolio is annualized and presented under the heading $\mathbb{E}[R^e_{t+1}]$. The annualized pricing error (α), its t-stat (t) and the two-tailed p-value (p) are calculated from a projection of portfolio returns onto the returns of the market factor ($f^{(\text{MKT})}_t$), size factor ($f^{(\text{SMB})}_t$), the value factor ($f^{(\text{HML})}_t$), the profitability factor ($f^{(\text{RMW})}_t$), and the investment factor ($f^{(\text{CMA})}_t$). Excess returns and pricing errors are expressed as percentage units. The GRS asset pricing (AP) test statistic is defined in Gibbons et al. (1989) and has the null hypothesis that the FF5 model correctly prices all decile portfolios. Newey-West standard errors are used to correct for heteroskedasticity and autocorrelation (using $T^{1/4} \cdot 0.75 \approx 2$ lags). Significance levels are specified as follows: * $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$.

Table A20
Fama-French Factor Construction

	Factor	Description
(1)	Market (MKT)	The weighted excess return $f_t^{(\text{MKT})}$ of all stocks available in each week t. The weight is defined by the prior week market capitalization (LME).
(2)	High Minus Low (HML)	Sort stocks (2×3) into 3 tertiles ("Value", "Neutral", "Growth") of book-to-market (BEME) and market cap (LME) separated at median ("Small", "Big"): $$f_t^{(\text{HML})} = \frac{1}{2}(\text{Small Value} + \text{Big Value}) - \frac{1}{2}(\text{Small Growth} + \text{Big Growth})$$
(3)	Robust Minus Weak (RMW)	Sort stocks (2×3) into 3 tertiles ("Weak", "Neutral", "Robust") of operating profit (OP) and market cap (LME) separated at median ("Small", "Big"): $$f_t^{(\text{RMW})} = \frac{1}{2}(\text{Small Robust} + \text{Big Robust}) - \frac{1}{2}(\text{Small Weak} + \text{Big Weak})$$
(4)	Conservative Minus Aggressive (CMA)	Sort stocks (2×3) into 3 tertiles ("Conserative", "Neutral", "Aggressive") of investment (Investment) and market cap (LME) separated at median ("Small", "Big"): $$f_t^{(\text{CMA})} = \frac{1}{2}(\text{Small Robust} + \text{Big Robust}) - \frac{1}{2}(\text{Small Weak} + \text{Big Weak})$$
(5)	Small Minus Big (SMB)	$$f_t^{(\text{SMB})} = \frac{1}{3}(\text{SMB}_{\text{BEME}} + \text{SMB}_{\text{OP}} + \text{SMB}_{\text{Investment}})$$ where SMB_{BEME} is defined as: $$\text{SMB}_{\text{BEME}} = (\text{Small Value} + \text{Small Neutral} + \text{Small Growth}) - (\text{Big Value} + \text{Big Neutral} + \text{Big Growth})$$ and SMB_{OP} and $\text{SMB}_{\text{Investment}}$ are defined analogously. In the FF3 model, $f_t^{(\text{SMB})}$ reduces to SMB_{BEME}.

Notes. For the HML, RMW, CMA, and SMB, the sorting of median-split and tertile sort of the characteristic is performed for the latest available characteristic value in June of any year. All factors are value-weighted within groups. The factor definitions are taken from Kenneth R. French's Dartmouth webpage (French, 2024)

AI-Disclosure Appendix

I have nothing to disclose.

Andreas Karlsson

Andreas Karlsson

May 13th, 2024

Comment on SSE extension to St. Gallen thesis

Andreas Karlsson (24419)

Summary

For my double degree with the University of St. Gallen I am required to make an extension to the St. Gallen version of the thesis. I have chosen to consolidate this extension and the St. Gallen thesis into a final SSE thesis. It makes more sense to consolidate the thesis in this way as it ends up saving a lot of space in the end. In particular, the results section of the SSE extension and the St. Gallen thesis are inextricably linked by the research questions I defined. This way, I can combine tables and figures for the St.Gallen thesis and the SSE extension for increased readability and for easier comparison. This is crucial because the models I consider in the St. Gallen thesis serve as benchmarks to be compared to the new models I develop for my SSE extension. Furthermore, as a future applicant to doctoral programs I need to be able to attach my thesis as a single readable document.

List of SSE extension amendments

- **Titlepage**:
  - New titlepage and abstract
- **Introduction**:
  - New first paragraph
  - Added section “Research Questions: Permanent-Transitory Decomposition”.
- **Literature**:
  - Added section “Literature: Permanent-Transitory Decomposition”.
- **Data and Empirical Work:**
  - Added section “Bond Characteristics”.
  - Added section “Permanent-Transitory Decomposition and GAN”.
  - Added new Figure II and Figure III
  - Added text on AJ-Beta models in section “Risk Loadings”. Added text describing on new research questions in section “Evaluation Metrics”. Added text on ensemble learning procedure and hyperparameter selection for SSE extension models.
- **Empirical Results for Nordic Equities**:
  - **IV.A**: Added SSE extension results to (main) Table I. Added paragraphs describing SSE extension results. Added new Panel B in Figure IV.
  - **IV.B**: Added new Table III. Added paragraphs describing SSE extension results.
  - **IV.C**: Added new Figure VII and Figure VIII. Added paragraphs describing SSE extension results.
  - **IV.D**: Added new Table VI. Added paragraphs describing SSE extension results.
  - **IV.E**: Added new Figure XII. Added paragraphs describing SSE extension results.
- **Discussion**:
  - Added paragraphs discussing SSE extension results.
- **Concluding Remarks**:
  - Rewrote paragraphs to include SSE extension.
- **Appendix and References**
  - Added SSE references to reference section.
  - **A1**. Added new Figure A1, Table A1, Figure A6, Figure A7, Figure A8.
  - **A2**. Added new Table A4, Table A5, Table A6
  - **A3**. Added new Figure A12, Figure A13
  - **A4**. Added new Algorithm 2
  - **A5**. Added Panel B in Figure A16. Table A9 (Part 1), Table A9 (Part 2), Table A10 (Part 1), Table A10 (Part 2),
  - **A6**. Added right column in Figure A17, Figure A28. Added Figure A18, Figure A19, Figure A20, Figure A21, Figure A22, Figure A23, Figure A27, Figure A29
  - **A7**. Added paragraph describing new regressions. Added Table A18, Table A19.
  - Added AI disclosure document.