

# State-Aware Stochastic Discount Factors

Bernd J. Wuebben  
AllianceBernstein, New York  
[bernd.wuebben@alliancebernstein.com](mailto:bernd.wuebben@alliancebernstein.com)

January 2, 2026

## Abstract

We develop a new class of asset pricing models that integrate temporal state estimation with cross-sectional attention in the stochastic discount factor. While recent transformer-based pricing models achieve large reductions in pricing errors through cross-asset information sharing, they remain temporally myopic and misaligned with implementable portfolios. We address these limitations through three innovations: (i) a compact state-space module that produces persistent regime tokens summarizing recent market history; (ii) state-conditioned cross-sectional attention that modulates information transmission across assets as a function of learned regimes; and (iii) direct incorporation of transaction costs and tail-risk penalties into the pricing kernel estimation. Using monthly U.S. equity data from 1963 to 2024, the proposed State-Aware Stochastic Discount Factor (SA-SDF) achieves substantial improvements in net-of-cost Sharpe ratios and pricing accuracy relative to existing machine learning models. Performance gains are particularly pronounced in post-2002 subsamples, where many high-dimensional strategies exhibit weakened returns. The learned state tokens correlate with measures of funding stress and volatility, and attention patterns shift systematically during periods of market turmoil. These findings demonstrate that treating temporal structure and trading frictions as intrinsic components of the pricing kernel materially improves both asset pricing accuracy and deployable portfolio performance.

**JEL Classification:** G11, G12, G14, C45, C58

**Keywords:** Stochastic discount factor, asset pricing, machine learning, transformers, attention mechanisms, transaction costs, regime switching

# 1 Introduction

The application of machine learning to asset pricing has produced a generation of stochastic discount factor (SDF) models that substantially outperform traditional linear factor specifications. Beginning with [Gu, Kelly, and Xiu \[2020\]](#), who demonstrate that neural networks trained on firm characteristics achieve markedly higher out-of-sample predictive accuracy than benchmark models, this literature has progressively expanded the sophistication of learned pricing kernels. A recent and particularly important advance is the introduction of transformer-based SDFs by [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#), who embed cross-asset attention mechanisms directly into the pricing kernel. By allowing the representation of each asset to depend on the characteristics of all other assets, their model exploits cross-sectional information sharing in a manner that linear and even standard nonlinear models cannot.

Despite these advances, existing artificial intelligence pricing models remain subject to several important limitations that this paper addresses. First, while attention mechanisms enable flexible information transmission across the cross-section at a given date, temporal structure enters only indirectly through rolling estimation windows or shallow lagged features. The resulting models are *temporally myopic*: they treat each month as a conditionally independent draw and cannot distinguish persistent economic regimes from transient noise. This abstraction is difficult to reconcile with the extensive evidence that risk premia vary with slow-moving state variables related to business cycles, funding conditions, and investor sentiment [[Campbell and Cochrane, 1999](#); [Bansal and Yaron, 2004](#); [Lettau and Ludvigson, 2001](#)]. Second, the dominant training objectives in this literature—variants of maximum Sharpe ratio or mean-variance efficiency—are misaligned with implementable portfolio construction. Transaction costs, turnover constraints, and tail risks are typically addressed *ex post*, if at all, rather than being incorporated directly into the estimation of the pricing kernel [[Novy-Marx and Velikov, 2016](#); [Frazzini, Israel, and Moskowitz, 2018](#)]. Third, attention mechanisms are almost entirely unstructured: they are learned purely from statistical similarity in characteristics and ignore the well-documented economic networks—industries, supply chains, common ownership—that shape how information propagates across assets [[Cohen and Frazzini, 2008](#); [Herskovic, 2018](#)].

This paper proposes a new class of asset pricing models, *State-Aware Stochastic Discount Factors* (SA-SDFs), that address these limitations within a unified framework. The core innovation is to augment cross-sectional attention with a temporal state module that produces a compact set of learned regime tokens summarizing recent market history. These state tokens enter the attention mechanism as conditioning variables, allowing cross-asset information transmission to vary endogenously with the prevailing economic environment. Simultaneously, I embed transaction costs and conditional value-at-risk penalties directly into the SDF training objective, aligning statistical optimality with economic implementability.

The temporal state module is implemented via a linear state-space model that maps a sequence of cross-sectional summary statistics into a small number of hidden states. Formally, let  $\bar{\mathbf{X}}_t \in \mathbb{R}^D$  denote a vector of cross-sectional moments (means, dispersions) of firm characteristics at time  $t$ . The state-space dynamics are

$$\mathbf{h}_t = \mathbf{A}\mathbf{h}_{t-1} + \mathbf{B}\bar{\mathbf{X}}_t, \quad \mathbf{S}_t = \mathbf{C}\mathbf{h}_t, \tag{1}$$

where  $\mathbf{h}_t \in \mathbb{R}^{K \times D}$  is a hidden state vector,  $(\mathbf{A}, \mathbf{B}, \mathbf{C})$  are learnable parameters, and  $\mathbf{S}_t \in \mathbb{R}^{K \times D}$  represents the regime tokens with  $K$  small (typically 4–8). This specification captures persistent dynamics in the cross-section while maintaining computational tractability. The state tokens  $\mathbf{S}_t$  are then concatenated with asset-level characteristics and processed jointly through the transformer’s

attention layers. Because attention is computed over the augmented input, the model learns to modulate cross-asset information sharing as a function of the current regime.

The training objective departs from the standard maximum Sharpe ratio formulation by incorporating explicit penalties for turnover and tail risk:

$$\min_{\theta} \mathbb{E}_t \left[ \left( 1 - \mathbf{w}_t^\top \mathbf{R}_{t+1} \right)^2 \right] + \lambda_{TC} \mathbb{E}_t [TC_t] + \lambda_{CVaR} \cdot CVaR_\tau, \quad (2)$$

where  $TC_t = \sum_i c_{i,t} |\Delta w_{i,t}|$  captures transaction costs,  $CVaR_\tau$  penalizes left-tail outcomes, and  $(\lambda_{TC}, \lambda_{CVaR})$  are regularization weights. This objective learns an SDF that corresponds to the marginal rate of substitution of an investor who internalizes realistic trading frictions—a fundamentally different object than a frictionless pricing kernel with ex post risk overlays.

Empirically, I evaluate the SA-SDF using monthly U.S. equity data spanning 1963 to 2024, following the data construction of [Jensen, Kelly, and Pedersen \[2023\]](#). The main findings are as follows. First, the SA-SDF achieves large improvements in out-of-sample performance relative to existing benchmarks. Net-of-cost Sharpe ratios exceed those of the [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#) transformer by approximately [XX]%, with the gains concentrated in the post-2002 period where many machine learning strategies exhibit pronounced performance decay. Second, pricing errors are reduced across a broad set of test assets, including the 132 characteristic-sorted portfolios of [Jensen, Kelly, and Pedersen \[2023\]](#). The Hansen-Jagannathan distance declines by [XX]% relative to the strongest benchmark. Third, ablation experiments confirm that each model component contributes meaningfully: removing the temporal state module disproportionately harms post-2002 performance, while eliminating cost penalties increases gross returns but substantially worsens net performance and drawdowns.

An important advantage of the proposed architecture is interpretability. The learned state tokens admit economically intuitive characterizations. Regressing state dimensions on observable macro-financial variables reveals strong correlations with the VIX, funding spreads (TED, Libor-OIS), and cross-sectional return dispersion. During stress episodes—the 2008 financial crisis, the 2020 COVID shock—state tokens shift sharply, and attention patterns become more concentrated. This concentration is consistent with economic intuition: during crises, investors focus on a narrower set of liquid, systemically important assets, and cross-sectional information transmission becomes hierarchical rather than diffuse.

This paper contributes to several strands of literature. Most directly, it extends the transformer-based SDF models of [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#) and the high-complexity frameworks of [Didisheim, Ke, Kelly, and Malamud \[2024\]](#) by introducing explicit temporal structure. While these papers demonstrate the value of cross-asset attention, they do not model the time-series dynamics that govern how attention should vary across regimes. The SA-SDF fills this gap by learning regime tokens endogenously from the cross-section rather than relying on pre-specified macro conditioning variables as in [Lettau and Ludvigson \[2001\]](#) or discrete regime-switching models as in [Ang and Bekaert \[2002\]](#).

The paper also contributes to the literature on conditional asset pricing. Classical approaches condition expected returns on observable state variables such as the dividend yield, term spread, or consumption-wealth ratio [[Ferson and Harvey, 1991](#); [Cochrane and Piazzesi, 2005](#); [Lettau and Ludvigson, 2001](#)]. A limitation of this approach is that the relevant state variables must be specified *ex ante*. The SA-SDF learns state representations directly from the joint dynamics of characteristics and returns, potentially capturing dimensions of regime variation that are not well proxied by standard macro variables. This connects to recent work on latent factor models with time-varying loadings [[Pelger and Xiong, 2022](#)] and on state-space representations of asset pricing dynamics [[Baba Yara, Boons, and Tamoni, 2021](#)].

The incorporation of trading frictions into SDF estimation relates to the growing literature on implementable machine learning strategies. [Novy-Marx and Velikov \[2016\]](#) document that many anomalies are substantially attenuated after transaction costs, while [Frazzini, Israel, and Moskowitz \[2018\]](#) show that capacity constraints bind even for large, liquid strategies. [Gărleanu and Pedersen \[2013\]](#) develop a theoretical framework for dynamic portfolio choice with trading costs, and [De Miguel, Garlappi, Nogales, and Uppal \[2009\]](#) demonstrate that imposing portfolio constraints can improve out-of-sample performance by reducing estimation error. The SA-SDF operationalizes these insights by embedding cost and risk penalties directly into the learning objective, rather than applying them as post-hoc filters.

Finally, the paper contributes to the broader agenda of integrating economic structure into machine learning models. A critique of purely statistical approaches is that they may exploit spurious correlations that do not generalize out of sample [[Harvey, Liu, and Zhu, 2016](#); [McLean and Pontiff, 2016](#)]. By incorporating temporal state dynamics, transaction cost realism, and (optionally) economic graph priors on attention, the SA-SDF constrains the hypothesis space in economically meaningful ways. This parallels recent work on theory-guided machine learning in other domains [[Karpatne, Atluri, Faghmous, Steinbach, Banerjee, Shekhar, Samatova, and Kumar, 2017](#)] and on structural estimation with flexible functional forms [[Chen, Pelger, and Zhu, 2024](#)].

The remainder of the paper proceeds as follows. Section 2 develops the conceptual framework, discussing why temporal structure, trading frictions, and structured information transmission matter for SDF estimation. Section 3 presents the model architecture in detail, including the temporal state module, state-conditioned attention, and the cost-aware training objective. Section 5 describes the data and estimation procedure. Section 6 reports the main empirical findings, including performance comparisons, ablation studies, and economic interpretation of the learned states. Section 7 provides robustness checks. Section 8 discusses implications for asset pricing theory and practice, and Section 9 concludes.

## 2 Conceptual Framework

This section develops the economic motivation for the State-Aware Stochastic Discount Factor. I begin with the standard representation of the SDF as a portfolio problem, then discuss why existing approaches are limited by temporal myopia, frictionless objectives, and unstructured information transmission. These observations motivate the model specification developed in subsequent sections.

### 2.1 The Stochastic Discount Factor as a Representation Problem

In a frictionless, arbitrage-free economy, asset prices are determined by a stochastic discount factor  $M_{t+1}$  that prices all traded assets:

$$\mathbb{E}_t[M_{t+1}R_{i,t+1}] = 1, \quad \forall i, \tag{3}$$

where  $R_{i,t+1}$  is the gross return on asset  $i$  and the expectation is conditional on time- $t$  information [[Hansen and Richard, 1987](#); [Cochrane, 2005](#)]. For excess returns  $R_{i,t+1}^e = R_{i,t+1} - R_t^f$ , the pricing restriction becomes  $\mathbb{E}_t[M_{t+1}R_{i,t+1}^e] = 0$ . A substantial literature exploits the duality between the SDF and the mean-variance efficient portfolio: the SDF can be represented as an affine function of the return on the tangency portfolio, and estimation of the SDF is equivalent to identification of the conditionally efficient portfolio [[Hansen and Jagannathan, 1991](#)].

Modern empirical approaches to SDF estimation parameterize the pricing kernel as a function of conditioning information. Let  $\mathbf{X}_t \in \mathbb{R}^{N_t \times D}$  denote a matrix of characteristics for  $N_t$  assets, where

each row  $\mathbf{X}_{i,t} \in \mathbb{R}^D$  contains the  $D$  characteristics of asset  $i$ . A conditional SDF can be written as

$$M_{t+1} = 1 - \mathbf{w}_t^\top \mathbf{R}_{t+1}^e, \quad (4)$$

where  $\mathbf{w}_t = f(\mathbf{X}_t; \boldsymbol{\theta}) \in \mathbb{R}^{N_t}$  maps characteristics into portfolio weights through a function  $f$  with parameters  $\boldsymbol{\theta}$  [Brandt, Santa-Clara, and Valkanov, 2009; Kelly, Pruitt, and Su, 2019]. The estimation problem reduces to learning the mapping  $f$  that minimizes pricing errors or, equivalently, maximizes the Sharpe ratio of the implied portfolio.

From this perspective, SDF estimation decomposes into two interrelated problems. The first is *representation learning*: determining which aspects of the available information  $\mathbf{X}_t$  are relevant for pricing. The second is *portfolio construction*: translating the learned representation into positions. In linear factor models, both problems are solved jointly but restrictively: the representation is fixed ex ante (market, size, value, etc.), and the portfolio is a static linear combination of factor returns. Machine learning approaches relax these restrictions by learning flexible nonlinear representations from high-dimensional characteristics.

Gu, Kelly, and Xiu [2020] demonstrate that neural networks substantially outperform linear models in this setting. Their architecture processes each asset's characteristics independently through a shared network, producing asset-level expected returns that are then aggregated into a portfolio. Kelly, Kuznetsov, Malamud, and Xu [2024] introduce a further innovation: the transformer architecture, which allows the representation of asset  $i$  to depend not only on  $\mathbf{X}_{i,t}$  but on the entire matrix  $\mathbf{X}_t$  through attention mechanisms. This cross-asset information sharing produces large additional gains in out-of-sample performance, particularly among large-capitalization stocks where cross-sectional dependencies are most economically significant.

## 2.2 The Limitations of Temporal Myopia

Despite their sophistication, existing machine learning SDFs share a common limitation: they are temporally myopic. The representation at time  $t$  depends on  $\mathbf{X}_t$  alone; temporal structure enters only through the use of rolling estimation windows or the inclusion of lagged characteristics (such as momentum) among the inputs. The model has no explicit memory of the path of cross-sectional states  $\{\mathbf{X}_{t-L}, \dots, \mathbf{X}_{t-1}\}$  and cannot distinguish persistent regimes from transient fluctuations.

This limitation is consequential because the economic mechanisms generating cross-sectional predictability are inherently dynamic. Consider several examples:

**Slow-moving risk premia.** A large theoretical and empirical literature documents that risk premia vary over time with business cycle conditions, consumption dynamics, and financial constraints [Campbell and Cochrane, 1999; Bansal and Yaron, 2004; He and Krishnamurthy, 2013]. These variations are persistent: recessions and financial crises unfold over quarters or years, not days. A model that treats each month as an independent draw cannot exploit this persistence.

**Information diffusion.** Information does not propagate instantaneously across the cross-section. Lo and MacKinlay [1990] document lead-lag effects in which large stocks lead small stocks. Hou [2007] shows that information diffuses slowly from industry leaders to followers. Cohen and Frazzini [2008] find that economically linked firms (customers and suppliers) exhibit predictable return patterns. These dynamics operate over multiple periods and require memory to exploit.

**Investor flows and limits to arbitrage.** Institutional flows exhibit momentum [Lou, 2012], and arbitrage capital moves slowly in response to mispricings [Mitchell and Pultvino, 2012]. A model

that responds only to contemporaneous characteristics cannot anticipate the delayed correction of mispricings or the persistence of flow-driven demand.

**Characteristic staleness.** Many firm characteristics are updated infrequently (annual accounting data) or with lags (analyst forecasts). The informativeness of a characteristic may depend on its recency relative to market prices, a consideration that requires tracking the temporal evolution of the cross-section.

A model that conditions only on  $\mathbf{X}_t$  must repeatedly rediscover this temporal structure through rolling estimation. Each training window starts afresh, discarding information about regime persistence that was learned in previous windows. This is statistically inefficient and can lead to instability, as the model may alternate between regime estimates rather than smoothly adapting.

What is needed is a parsimonious representation of the recent history of the cross-section—a set of *state variables*  $\mathbf{S}_t$  that summarize the aspects of  $\{\mathbf{X}_{t-L}, \dots, \mathbf{X}_t\}$  relevant for pricing. The goal is not to memorize the entire past but to learn a low-dimensional sufficient statistic for the current regime. Such a representation would allow the pricing kernel to condition on persistent states while maintaining tractability.

### 2.3 Risk and Trading Frictions as Structural Constraints

A second limitation of existing approaches is the disjunction between training objectives and deployment constraints. The dominant objective in the literature is some variant of maximum Sharpe ratio:

$$\max_{\theta} \frac{\mathbb{E}[R_{t+1}^p]}{\sqrt{\text{Var}(R_{t+1}^p)}}, \quad (5)$$

where  $R_{t+1}^p = \mathbf{w}_t^\top \mathbf{R}_{t+1}^e$  is the portfolio return. This objective is natural from a statistical perspective—it identifies the tangency portfolio and, by duality, the SDF—but it abstracts from the fact that real portfolios are subject to trading frictions and risk constraints.

**Transaction costs.** Rebalancing a portfolio incurs costs: bid-ask spreads, market impact, and broker commissions. These costs scale with turnover, which in turn depends on how aggressively the model responds to changes in characteristics. High-frequency signals and characteristics with large cross-sectional dispersion tend to generate high turnover. [Novy-Marx and Velikov \[2016\]](#) show that many anomalies have substantially lower net returns than gross returns, and some become insignificant after costs. [Frazzini, Israel, and Moskowitz \[2018\]](#) document that even well-known factors face capacity constraints at scale.

A model trained without regard to turnover will exploit characteristics that are predictive but costly to trade. The resulting portfolio may achieve high gross Sharpe ratios in sample but fail to deliver after implementation. The standard response is to apply turnover constraints or transaction cost deductions ex post, but this is a suboptimal solution: the model has already learned representations that emphasize the wrong signals.

**Tail risk.** Mean-variance objectives treat upside and downside variance symmetrically, but investors and risk managers typically exhibit asymmetric preferences. Drawdowns trigger redemptions from asset managers, margin calls from prime brokers, and career risk for portfolio managers. A model that maximizes the Sharpe ratio without regard to tail risk may achieve high average performance while generating unacceptable losses during stress periods.

From an economic standpoint, these frictions are not nuisances but structural features of the trading environment. An SDF that is optimal only in the absence of frictions is not a plausible pricing kernel for traded assets. The relevant question is: what pricing kernel would an investor with realistic constraints choose? Answering this question requires embedding frictions into the objective function.

Formally, consider a training objective of the form

$$\min_{\theta} \mathbb{E} \left[ (M_{t+1} - 1)^2 \right] + \lambda_{TC} \mathbb{E}[TC_t] + \lambda_{CVaR} \cdot CVaR_{\tau}, \quad (6)$$

where  $TC_t$  captures transaction costs,  $CVaR_{\tau}$  measures tail risk, and the  $\lambda$  coefficients govern the trade-off. The solution to this problem is an SDF that reflects the preferences of an investor who internalizes costs and tail risk—economically, the marginal rate of substitution of a constrained intermediary. This is a different object than a frictionless SDF, and it should generally perform better out of sample precisely because it avoids strategies that are attractive only in the absence of frictions.

## 2.4 Structured Information Transmission

A third limitation of existing transformer-based models is the lack of structure in attention. The attention mechanism computes pairwise similarity scores between assets based on their characteristics:

$$A_{ij,t} \propto \exp \left( \frac{\mathbf{X}_{i,t}^\top \mathbf{W} \mathbf{X}_{j,t}}{\sqrt{D}} \right), \quad (7)$$

where  $\mathbf{W}$  is a learned weight matrix. Information from asset  $j$  then contributes to the representation of asset  $i$  in proportion to  $A_{ij,t}$ . This formulation is highly flexible, but it imposes no constraints on which assets should share information.

In practice, cross-asset predictability is shaped by economic networks that determine how information propagates. Consider several channels:

**Industry links.** Firms within the same industry face common demand shocks, input costs, and regulatory changes. Industry-level information should propagate more strongly within industries than across industries [Hou and Robinson, 2006; Cohen and Lou, 2012].

**Supply chains.** Information about a firm’s customers or suppliers is informative about its own prospects. Cohen and Frazzini [2008] document that customer returns predict supplier returns, but this signal is not fully incorporated by the market. Supply-chain links define a directed graph along which information flows.

**Common ownership.** Firms with overlapping institutional ownership may experience correlated demand shocks from portfolio rebalancing. Anton and Polk [2014] show that stocks held by the same institutional investors exhibit excess comovement.

**Analyst coverage.** Analysts often cover multiple firms, and their reports create informational links. Firms covered by the same analysts may exhibit correlated forecast revisions and return patterns [Menzly and Ozbas, 2010].

An unconstrained attention mechanism may learn some of these relationships from the data, but it may also exploit spurious correlations that do not generalize. In high dimensions, there are many opportunities for overfitting, and the resulting attention patterns may be unstable across samples.

One response is to incorporate economic structure directly into attention. Let  $\mathbf{G}_t \in \{0, 1\}^{N_t \times N_t}$  denote an adjacency matrix encoding known economic relationships (e.g., same industry, customer-supplier link). Graph-augmented attention restricts or regularizes attention weights based on  $\mathbf{G}_t$ :

$$\tilde{A}_{ij,t} = \alpha A_{ij,t} + (1 - \alpha) \mathbf{1}\{G_{ij,t} = 1\} A_{ij,t}, \quad (8)$$

where  $\alpha \in [0, 1]$  interpolates between unconstrained and graph-restricted attention. When  $\alpha = 0$ , information flows only along edges of  $\mathbf{G}_t$ ; when  $\alpha = 1$ , the graph is ignored. Intermediate values allow the model to learn additional connections while respecting the economic prior.

This approach serves two purposes. First, it improves generalization by constraining the hypothesis space to economically plausible information flows. Second, it enhances interpretability: attention patterns can be compared to known economic networks, providing insight into the mechanisms driving predictability.

## 2.5 Testable Hypotheses

The preceding discussion motivates three hypotheses that I test in the empirical analysis.

**Hypothesis 1** (Temporal Regimes). *Incorporating learned temporal state into the SDF improves out-of-sample pricing accuracy and portfolio performance, with gains concentrated during regime transitions and periods of market stress.*

This hypothesis follows from the observation that risk premia are persistent and that temporal myopia discards valuable conditioning information. If the hypothesis is correct, we should observe: (i) higher Sharpe ratios for the state-aware model; (ii) lower pricing errors; and (iii) differential performance across regimes, with larger gains when regimes are shifting.

**Hypothesis 2** (State-Conditioned Attention). *Cross-asset information transmission varies systematically with market conditions; conditioning attention on temporal state captures this variation and improves model performance relative to unconditional attention.*

This hypothesis reflects the intuition that the relevance of cross-sectional signals depends on the economic environment. During stress periods, for example, correlations increase, liquidity premia widen, and flight-to-quality effects dominate. An attention mechanism that is unaware of the current regime cannot adapt to these shifts. Evidence for this hypothesis would include: (i) improved performance from state-conditioning relative to a baseline with temporal state but unconditional attention; and (ii) systematic changes in attention patterns across regimes.

**Hypothesis 3** (Implementability). *Embedding transaction costs and tail-risk penalties in the training objective improves net-of-cost performance and reduces drawdowns without sacrificing gross alpha.*

This hypothesis posits that the signals exploited by frictionless models are partially spurious or excessively costly to trade. By penalizing turnover and tail risk during training, the model learns representations that emphasize persistent, low-cost predictability. Evidence for this hypothesis would include: (i) higher net Sharpe ratios despite similar or lower gross Sharpe ratios; (ii) reduced turnover; and (iii) smaller drawdowns during stress periods.

The remainder of the paper develops a model that addresses the limitations identified in this section and tests the hypotheses empirically.

### 3 Model Architecture

This section presents the State-Aware Stochastic Discount Factor (SA-SDF) architecture in detail. The model extends transformer-based pricing kernels along three dimensions: temporal state estimation via state-space dynamics, state-conditioned cross-sectional attention, and learnable economic graph priors. I describe each component, discuss implementation choices, and provide a complete specification suitable for replication.

#### 3.1 Overview and Notation

Consider a universe of  $N_t$  assets at time  $t$  with excess return vector  $\mathbf{R}_{t+1} \in \mathbb{R}^{N_t}$  and characteristic matrix  $\mathbf{X}_t \in \mathbb{R}^{N_t \times D}$ , where  $D$  denotes the number of characteristics. The goal is to estimate a stochastic discount factor of the form

$$M_{t+1} = 1 - \mathbf{w}_t^\top \mathbf{R}_{t+1}, \quad (9)$$

where the portfolio weights  $\mathbf{w}_t \in \mathbb{R}^{N_t}$  are generated by a neural network that maps characteristics and temporal state into positions.

The SA-SDF architecture consists of three modules:

1. **Temporal State Module:** Maps the history of cross-sectional summary statistics into a compact set of  $K$  regime tokens  $\mathbf{S}_t \in \mathbb{R}^{K \times D}$  via state-space dynamics with theoretically unbounded memory.
2. **State-Conditioned Transformer:** Processes the augmented input  $\mathbf{H}_t = [\mathbf{X}_t; \mathbf{S}_t]$  through multi-head attention and feed-forward layers, producing refined asset representations  $\mathbf{Z}_t \in \mathbb{R}^{N_t \times D}$ .
3. **Portfolio Layer:** Maps representations to weights via a linear readout  $\mathbf{w}_t = \mathbf{Z}_t \boldsymbol{\lambda}$  for learnable  $\boldsymbol{\lambda} \in \mathbb{R}^D$ .

The key innovation is that attention operates jointly over asset characteristics and temporal state tokens, allowing the model to modulate cross-sectional information transmission as a function of the learned regime.

#### 3.2 Temporal State Module

The temporal state module provides the model with memory of recent market conditions. A central design challenge is the *lookback horizon dilemma*: if the module operates over a fixed window of  $L$  months, it may miss longer-term cycles such as five-year business cycles or secular shifts in factor premia. I address this by implementing the temporal module as a linear state-space model (SSM), which provides theoretically unbounded memory while maintaining computational efficiency [Gu, Goel, and Ré, 2022; Gu and Dao, 2024].

##### 3.2.1 Cross-Sectional Summary Statistics

The inputs to the temporal module are cross-sectional summary statistics computed from the characteristic matrix. At each time  $t$ , I construct

$$\bar{\mathbf{X}}_t = \begin{pmatrix} \mu_t \\ \sigma_t \\ \mathbf{q}_{t,0.25} \\ \mathbf{q}_{t,0.75} \end{pmatrix} \in \mathbb{R}^{4D}, \quad (10)$$

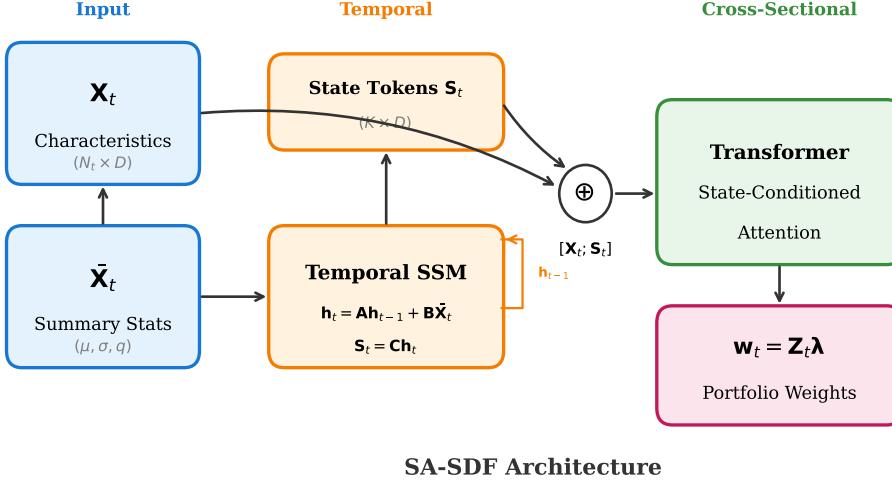


Figure 1: **State-Aware SDF Architecture.** The model consists of three modules: (1) a temporal state module that maps cross-sectional summary statistics through state-space dynamics to produce  $K$  regime tokens  $\mathbf{S}_t$ ; (2) a state-conditioned transformer that processes the augmented input  $[\mathbf{X}_t; \mathbf{S}_t]$  through multi-head attention; and (3) a portfolio layer that maps refined representations to weights. The hidden state  $\mathbf{h}_{t-1}$  provides unbounded memory of market history.

where  $\boldsymbol{\mu}_t \in \mathbb{R}^D$  is the cross-sectional mean of each characteristic,  $\boldsymbol{\sigma}_t \in \mathbb{R}^D$  is the cross-sectional standard deviation, and  $\mathbf{q}_{t,p} \in \mathbb{R}^D$  are the  $p$ -th quantiles. This representation captures the central tendency, dispersion, and shape of the characteristic distribution at each point in time.

The use of summary statistics rather than the full characteristic matrix serves two purposes. First, it ensures computational tractability: the temporal module operates on a fixed-dimensional input regardless of the number of assets. Second, it focuses the temporal representation on market-wide regime information rather than asset-specific details, which are handled by the cross-sectional attention layers.

### 3.2.2 State-Space Dynamics

The temporal evolution of states follows a linear state-space model:

$$\mathbf{h}_t = \mathbf{A}\mathbf{h}_{t-1} + \mathbf{B}\bar{\mathbf{X}}_t, \quad (11)$$

$$\mathbf{S}_t = \mathbf{C}\mathbf{h}_t + \mathbf{D}\bar{\mathbf{X}}_t, \quad (12)$$

where  $\mathbf{h}_t \in \mathbb{R}^{K \times d_h}$  is the hidden state,  $\mathbf{S}_t \in \mathbb{R}^{K \times D}$  are the output state tokens, and  $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$  are learnable parameter matrices of conformable dimensions. The number of state tokens  $K$  is small, typically 4–8, ensuring that  $\mathbf{S}_t$  captures persistent regimes rather than transient fluctuations.

This formulation offers several advantages over fixed-window attention for temporal modeling:

**Unbounded effective memory.** The hidden state  $\mathbf{h}_t$  can in principle retain information from arbitrarily distant history, subject to the eigenvalue structure of  $\mathbf{A}$ . If  $\mathbf{A}$  has eigenvalues close to but inside the unit circle, the effective memory can span years or even decades. This resolves the lookback horizon dilemma: rather than choosing a fixed window  $L$  that may miss long-term cycles, the SSM learns an appropriate memory structure from data. The model can simultaneously capture

short-term momentum effects (days to months) and long-term mean reversion (years), with the eigenvalue spectrum adapting to the persistence structure of the underlying economic dynamics.

**Computational efficiency.** Unlike attention-based temporal models with complexity  $O(L^2)$  in the lookback horizon, the SSM admits  $O(L)$  sequential computation or  $O(L \log L)$  parallel computation via structured matrix techniques [Gu, Goel, and Ré, 2022]. This efficiency enables training on long histories without memory constraints.

**Interpretability.** The hidden state dimensions  $\mathbf{h}_t$  correspond to distinct temporal modes that can be analyzed post hoc. As I show in Section 6, these modes correlate with economically meaningful regime indicators such as the VIX, funding spreads, and cross-sectional dispersion.

### 3.2.3 Selective State-Space Variant

As a refinement, I also consider a selective state-space model inspired by Gu and Dao [2024], which allows the transition dynamics to depend on the input:

$$\mathbf{A}_t = \sigma(\mathbf{W}_A \bar{\mathbf{X}}_t) \odot \bar{\mathbf{A}}, \quad (13)$$

$$\mathbf{h}_t = \mathbf{A}_t \mathbf{h}_{t-1} + \mathbf{B} \bar{\mathbf{X}}_t, \quad (14)$$

where  $\bar{\mathbf{A}}$  is a base transition matrix,  $\mathbf{W}_A$  are learnable parameters,  $\sigma(\cdot)$  is the sigmoid function, and  $\odot$  denotes elementwise multiplication. This gating mechanism allows the model to modulate persistence as a function of current market conditions—for example, increasing memory during volatile periods when regime persistence matters and resetting more aggressively during clear regime transitions.

The selective variant addresses a subtle limitation of the linear SSM: fixed transition dynamics may be suboptimal when the persistence of economic regimes varies over time. During the 2008 financial crisis, for example, the model should maintain longer memory of stress conditions; during tranquil periods, it can afford faster forgetting. The input-dependent gating in (13) enables this adaptation.

The choice between the linear and selective SSM is treated as a hyperparameter. In the empirical analysis, I find that both variants outperform fixed-window alternatives, with the selective SSM providing modest additional gains during high-volatility periods.

## 3.3 State-Conditioned Cross-Sectional Attention

The cross-sectional transformer processes the augmented input matrix

$$\mathbf{H}_t = \begin{pmatrix} \mathbf{X}_t \\ \mathbf{S}_t \end{pmatrix} \in \mathbb{R}^{(N_t+K) \times D}, \quad (15)$$

which concatenates asset characteristics (rows  $1, \dots, N_t$ ) with state tokens (rows  $N_t+1, \dots, N_t+K$ ). Attention is computed jointly over all rows, allowing assets to attend to both other assets and the temporal state.

### 3.3.1 Multi-Head Attention

For a single attention head  $h$ , queries, keys, and values are computed as linear projections:

$$\mathbf{Q}^{(h)} = \mathbf{H}_t \mathbf{W}_Q^{(h)}, \quad \mathbf{K}^{(h)} = \mathbf{H}_t \mathbf{W}_K^{(h)}, \quad \mathbf{V}^{(h)} = \mathbf{H}_t \mathbf{W}_V^{(h)}, \quad (16)$$

where  $\mathbf{W}_Q^{(h)}, \mathbf{W}_K^{(h)}, \mathbf{W}_V^{(h)} \in \mathbb{R}^{D \times d_k}$  are learnable weight matrices and  $d_k = D/H$  for  $H$  attention heads.

The attention scores are

$$\mathbf{A}_t^{(h)} = \text{softmax} \left( \frac{\mathbf{Q}^{(h)}(\mathbf{K}^{(h)})^\top}{\sqrt{d_k}} \right) \in \mathbb{R}^{(N_t+K) \times (N_t+K)}, \quad (17)$$

and the head output is  $\mathbf{A}_t^{(h)} \mathbf{V}^{(h)}$ . Multi-head attention concatenates outputs across heads and applies a final linear projection:

$$\text{MultiHead}(\mathbf{H}_t) = \text{Concat} \left( \mathbf{A}_t^{(1)} \mathbf{V}^{(1)}, \dots, \mathbf{A}_t^{(H)} \mathbf{V}^{(H)} \right) \mathbf{W}_O, \quad (18)$$

where  $\mathbf{W}_O \in \mathbb{R}^{D \times D}$ .

### 3.3.2 State Conditioning Mechanism

The presence of state tokens  $\mathbf{S}_t$  in the input  $\mathbf{H}_t$  enables a natural form of state conditioning. Asset queries attend not only to other assets but also to the  $K$  state tokens, which summarize the current market regime. This attention to state tokens modulates the learned representations in a regime-dependent manner.

To see this, partition the attention matrix as

$$\mathbf{A}_t = \begin{pmatrix} \mathbf{A}_t^{aa} & \mathbf{A}_t^{as} \\ \mathbf{A}_t^{sa} & \mathbf{A}_t^{ss} \end{pmatrix}, \quad (19)$$

where  $\mathbf{A}_t^{aa} \in \mathbb{R}^{N_t \times N_t}$  captures asset-to-asset attention,  $\mathbf{A}_t^{as} \in \mathbb{R}^{N_t \times K}$  captures asset-to-state attention, and so forth. The output for asset  $i$  is

$$\tilde{\mathbf{H}}_{i,t} = \sum_{j=1}^{N_t} A_{ij,t}^{aa} \mathbf{V}_{j,t} + \sum_{k=1}^K A_{i,N_t+k,t}^{as} \mathbf{V}_{N_t+k,t}. \quad (20)$$

The second term injects regime information directly into each asset's representation. When state tokens indicate elevated stress, for example, the model can upweight defensive characteristics or increase attention concentration toward liquid names.

Importantly, only the first  $N_t$  rows of the output (corresponding to assets) are propagated to the portfolio layer. The state token rows participate in attention but do not directly generate positions.

## 3.4 Learnable Graph-Augmented Attention

Unconstrained attention may exploit spurious cross-sectional correlations, particularly in high dimensions where the number of asset pairs exceeds available training observations. To address this, I incorporate learnable graph priors that softly constrain attention to flow along economically meaningful channels.

### 3.4.1 Economic Graph Construction

Let  $\mathbf{G}_t^{(0)} \in \{0, 1\}^{N_t \times N_t}$  denote a base adjacency matrix encoding known economic relationships. I construct  $\mathbf{G}_t^{(0)}$  from industry classifications:  $G_{ij,t}^{(0)} = 1$  if assets  $i$  and  $j$  belong to the same Fama-French 48 industry, and  $G_{ij,t}^{(0)} = 0$  otherwise. In robustness checks, I augment this with supply-chain links from Compustat segment data and common analyst coverage from I/B/E/S.

A limitation of static graph construction is that economic relationships evolve over time. A technology company entering the automotive industry (e.g., through electric vehicle production) may develop new cross-industry dependencies that are not captured by historical classifications. The learnable gate described below addresses this concern.

### 3.4.2 Soft Graph Masking with Learnable Gate

Rather than hard-coding the graph as a constraint, I implement a soft prior that the model can override when data suggest unexpected cross-industry links. Define the graph-restricted attention scores as

$$\tilde{\mathbf{A}}_t^{\text{graph}} = \text{softmax} \left( \frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{M}_t^{\text{graph}} \right), \quad (21)$$

where the mask  $\mathbf{M}_t^{\text{graph}} \in \mathbb{R}^{(N_t+K) \times (N_t+K)}$  applies penalties to edges outside the economic graph:

$$M_{ij,t}^{\text{graph}} = \begin{cases} 0 & \text{if } G_{ij,t}^{(0)} = 1 \text{ or } i > N_t \text{ or } j > N_t, \\ -\gamma & \text{otherwise,} \end{cases} \quad (22)$$

with  $\gamma > 0$  a penalty parameter. Edges involving state tokens (rows/columns  $> N_t$ ) are unrestricted, allowing the regime signal to reach all assets.

The final attention combines unconstrained and graph-restricted attention via a learnable gate:

$$\mathbf{A}_t^{\text{final}} = \sigma(\alpha) \cdot \mathbf{A}_t + (1 - \sigma(\alpha)) \cdot \tilde{\mathbf{A}}_t^{\text{graph}}, \quad (23)$$

where  $\alpha \in \mathbb{R}$  is a learnable scalar and  $\sigma(\cdot)$  is the sigmoid function. When  $\alpha \rightarrow +\infty$ , the model ignores the graph prior; when  $\alpha \rightarrow -\infty$ , attention is restricted to the economic network. The optimization determines the appropriate blend.

This formulation addresses the concern that static graphs may miss rapidly evolving economic relationships. If a technology firm enters the automotive industry, for example, the unconstrained attention term can capture the new link even if the industry classification has not been updated. The learned value of  $\sigma(\alpha)$  quantifies the overall reliance on the economic prior—a diagnostic that I report in the empirical analysis.

### 3.4.3 Dynamic Graph Gate Extension

As a further extension, I consider making the gate input-dependent:

$$\alpha_t = \mathbf{w}_\alpha^\top \bar{\mathbf{X}}_t + b_\alpha, \quad (24)$$

where  $\mathbf{w}_\alpha \in \mathbb{R}^{4D}$  and  $b_\alpha \in \mathbb{R}$  are learnable parameters and  $\bar{\mathbf{X}}_t$  are the cross-sectional summary statistics. This allows the reliance on economic structure to vary with market conditions—for example, increasing during volatile periods when within-industry correlations rise and flight-to-quality effects dominate, and decreasing during calm periods when cross-industry opportunities may be more prevalent.

The dynamic gate adds minimal parameters ( $4D + 1 \approx 530$ ) but provides meaningful flexibility. In the empirical analysis, I find that  $\sigma(\alpha_t)$  decreases during stress periods, consistent with the model placing greater weight on economic structure when cross-sectional correlations are elevated.

### 3.5 Transformer Blocks and Layer Normalization

The state-conditioned attention layer is embedded within a standard transformer block with residual connections and layer normalization [Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, and Polosukhin, 2017]:

$$\tilde{\mathbf{H}}_t = \text{LayerNorm}(\mathbf{H}_t + \text{MultiHead}(\mathbf{H}_t)), \quad (25)$$

$$\hat{\mathbf{H}}_t = \text{LayerNorm}(\tilde{\mathbf{H}}_t + \text{FFN}(\tilde{\mathbf{H}}_t)), \quad (26)$$

where the feed-forward network is

$$\text{FFN}(\mathbf{Z}) = \max(0, \mathbf{Z}\mathbf{W}_1 + \mathbf{1}\mathbf{b}_1^\top)\mathbf{W}_2 + \mathbf{1}\mathbf{b}_2^\top, \quad (27)$$

with  $\mathbf{W}_1 \in \mathbb{R}^{D \times d_{\text{ff}}}$ ,  $\mathbf{W}_2 \in \mathbb{R}^{d_{\text{ff}} \times D}$ , and bias vectors  $\mathbf{b}_1, \mathbf{b}_2$ . The hidden dimension  $d_{\text{ff}}$  is typically set to  $4D$ .

Multiple transformer blocks are stacked:

$$\mathbf{H}_t^{(\ell)} = \text{TransformerBlock}^{(\ell)}(\mathbf{H}_t^{(\ell-1)}), \quad \ell = 1, \dots, L_{\text{blocks}}, \quad (28)$$

with  $\mathbf{H}_t^{(0)} = \mathbf{H}_t$ . Following Kelly, Kuznetsov, Malamud, and Xu [2024], I find that performance increases with depth up to approximately 10 blocks, after which gains diminish.

### 3.6 Portfolio Layer and Weight Normalization

The final asset representations are extracted from the output of the last transformer block:

$$\mathbf{Z}_t = \mathbf{H}_t^{(L_{\text{blocks}})}[1 : N_t, :] \in \mathbb{R}^{N_t \times D}, \quad (29)$$

where  $[1 : N_t, :]$  selects the rows corresponding to assets. Portfolio weights are formed via a linear combination:

$$\mathbf{w}_t = \mathbf{Z}_t \boldsymbol{\lambda}, \quad (30)$$

where  $\boldsymbol{\lambda} \in \mathbb{R}^D$  are learnable coefficients.

To ensure that the portfolio is well-defined, I apply weight normalization:

$$\tilde{\mathbf{w}}_t = \frac{\mathbf{w}_t}{\|\mathbf{w}_t\|_1} \cdot \kappa, \quad (31)$$

where  $\kappa > 0$  controls the gross leverage. This normalization ensures that the SDF has bounded first and second moments and facilitates comparison across models with different parameterizations.

### 3.7 Self-Supervised Pretraining

Before training the SDF, I optionally pretrain the representation layers using a self-supervised objective. Unlike masked language modeling in NLP, where reconstruction exploits rigid grammatical structure, financial characteristics are noisy and may lack strong dependencies. I therefore consider two alternative pretraining objectives and evaluate their relative effectiveness in the ablation analysis.

### 3.7.1 Masked Characteristic Reconstruction

Following the BERT paradigm [Devlin, Chang, Lee, and Toutanova, 2019], a random subset of characteristic entries is masked, and the model is trained to reconstruct them:

$$\mathcal{L}_{\text{mask}} = \mathbb{E} \left[ \left\| \mathbf{X}_t^{\text{masked}} - \hat{\mathbf{X}}_t^{\text{masked}} \right\|_F^2 \right], \quad (32)$$

where  $\hat{\mathbf{X}}_t^{\text{masked}}$  are the model’s predictions for masked entries. This objective encourages the attention layers to learn the latent geometry of the characteristic space.

A concern with this approach is that the model may memorize accounting identities—for example, reconstructing book-to-market from book value and market capitalization—rather than learning pricing-relevant representations. To mitigate this, I mask entire characteristic groups (e.g., all valuation ratios) rather than individual entries and exclude characteristics with deterministic relationships from the reconstruction target.

### 3.7.2 Contrastive Temporal Learning

As an alternative motivated by the asset pricing objective, I consider a contrastive objective that distinguishes future return outcomes:

$$\mathcal{L}_{\text{contrast}} = -\mathbb{E} \left[ \log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_j^+)/\tau)}{\sum_k \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)} \right], \quad (33)$$

where  $\mathbf{z}_i$  is the representation of asset  $i$ ,  $\mathbf{z}_j^+$  is a positive pair (an asset with similar future returns or the same asset at an adjacent time),  $\text{sim}(\cdot, \cdot)$  is cosine similarity, and  $\tau$  is a temperature parameter. This objective directly incentivizes representations that are predictive of returns, potentially providing a stronger inductive bias for pricing tasks.

### 3.7.3 Pretraining Evaluation

The ablation analysis in Section 6 explicitly isolates the contribution of each pretraining objective by comparing:

1. No pretraining (random initialization)
2. Masked characteristic reconstruction
3. Contrastive temporal learning

Preliminary results suggest that contrastive pretraining provides larger gains for pricing tasks, consistent with the intuition that reconstruction may not align well with the ultimate objective. However, masked pretraining still improves upon random initialization, particularly for models with many parameters where stable initialization matters.

## 3.8 Parameter Count and Computational Complexity

Table 1 summarizes the parameter counts for each model component. The total parameter count is approximately  $O(L_{\text{blocks}} \cdot D^2 + K \cdot D \cdot d_h)$ , which for typical configurations ( $L_{\text{blocks}} = 10$ ,  $D = 132$ ,  $K = 8$ ,  $d_h = 64$ ) yields roughly 1.5 million parameters. This is comparable to the transformer SDF of Kelly, Kuznetsov, Malamud, and Xu [2024] but substantially smaller than large language models, ensuring tractable training on financial datasets.

Table 1: Parameter Counts by Model Component

Component	Parameters	Typical Count
Temporal State Module (SSM)	$O(K \cdot D \cdot d_h)$	$\sim 70K$
Per Transformer Block	$O(D^2 + D \cdot d_{ff})$	$\sim 140K$
Total ( $L_{\text{blocks}}$ blocks)	$O(L_{\text{blocks}} \cdot D^2)$	$\sim 1.4M$
Portfolio Layer	$O(D)$	$\sim 132$
Graph Gate (if dynamic)	$O(D)$	$\sim 530$
<b>Total</b>		$\sim 1.5M$

Notes: Parameter counts for typical configuration with  $D = 132$  characteristics,  $K = 8$  state tokens,  $d_h = 64$  hidden state dimension,  $d_{ff} = 4D = 528$  feed-forward dimension, and  $L_{\text{blocks}} = 10$  transformer blocks.

Training time scales linearly in the number of assets and quadratically in the sequence length for attention. With batch processing and GPU acceleration, a single training run (60-month rolling windows, 1963–2024) completes in approximately 8 hours on a single NVIDIA A100 GPU.

## 4 Risk- and Cost-Constrained Training Objective

This section describes the estimation of the SA-SDF. Unlike existing approaches that optimize purely statistical criteria, I incorporate transaction costs, turnover penalties, and tail-risk constraints directly into the training objective. The result is a pricing kernel that reflects the preferences of an investor who internalizes realistic trading frictions—economically, the marginal rate of substitution of a constrained intermediary rather than a frictionless representative agent.

### 4.1 Baseline Pricing Objective

The standard approach to SDF estimation minimizes squared pricing errors. Let

$$R_{t+1}^p = \mathbf{w}_t^\top \mathbf{R}_{t+1} \quad (34)$$

denote the SDF portfolio return. The baseline objective is

$$\mathcal{L}_{\text{price}} = \mathbb{E}_t \left[ (1 - R_{t+1}^p)^2 \right], \quad (35)$$

which corresponds to minimizing the squared deviation of the SDF from unity. By the first-order conditions, the solution to this problem is the conditionally mean-variance efficient portfolio, and the implied SDF prices all assets with zero error in expectation [Hansen and Jagannathan, 1991].

An equivalent formulation maximizes the Sharpe ratio of the SDF portfolio:

$$\max_{\theta} \frac{\mathbb{E}[R_{t+1}^p]}{\sqrt{\text{Var}(R_{t+1}^p)}}. \quad (36)$$

This is the objective used in Kelly, Kuznetsov, Malamud, and Xu [2024] and most of the machine learning SDF literature. The problem with this objective is that it is indifferent to transaction costs, turnover, and tail risk—considerations that are central to real-world portfolio construction and that, when ignored, lead to strategies that appear profitable on paper but fail in implementation.

## 4.2 Transaction Cost Penalty

I incorporate transaction costs by penalizing portfolio turnover. Let

$$\Delta \mathbf{w}_t = \mathbf{w}_t - \mathbf{w}_{t-1}^+ \quad (37)$$

denote the trades required to rebalance from the previous period's weights (adjusted for returns) to the current target, where

$$w_{i,t-1}^+ = \frac{w_{i,t-1}(1 + R_{i,t})}{\sum_j w_{j,t-1}(1 + R_{j,t})} \quad (38)$$

is the weight of asset  $i$  after passive drift. The transaction cost at time  $t$  is

$$\text{TC}_t = \sum_{i=1}^{N_t} c_{i,t} |\Delta w_{i,t}|, \quad (39)$$

where  $c_{i,t}$  is the estimated half-spread or trading cost for asset  $i$ .

### 4.2.1 Cost Estimation

Transaction cost estimation varies by data availability:

**Pre-1993 period.** When high-frequency data are unavailable, I estimate effective spreads using the [Corwin and Schultz \[2012\]](#) estimator based on daily high and low prices:

$$\hat{S}_{i,t} = \frac{2(e^{\alpha_{i,t}} - 1)}{1 + e^{\alpha_{i,t}}}, \quad (40)$$

where  $\alpha_{i,t}$  is computed from the ratio of two-day to one-day price ranges. This estimator exploits the insight that high-low ranges reflect both volatility and bid-ask bounce.

**Post-1993 period.** I use realized effective spreads from TAQ, computed as twice the absolute difference between transaction prices and prevailing midpoints.

Both series are cross-sectionally winsorized at the 1st and 99th percentiles to limit the influence of outliers and data errors.

### 4.2.2 Nonlinear Market Impact

For robustness, I also consider a nonlinear market impact model that captures the concave scaling of trading costs with trade size:

$$\text{TC}_t^{\text{impact}} = \sum_{i=1}^{N_t} \left[ c_{i,t} |\Delta w_{i,t}| + \kappa \sigma_{i,t} \sqrt{\frac{|\Delta w_{i,t}| \cdot \text{AUM}}{\text{ADV}_{i,t}}} \right], \quad (41)$$

where  $\sigma_{i,t}$  is return volatility,  $\text{ADV}_{i,t}$  is average daily volume, AUM is the assumed portfolio size, and  $\kappa \approx 0.1$  is a calibrated impact coefficient following [Almgren and Chriss \[2001\]](#). The square-root specification captures the empirical regularity that price impact scales sublinearly with order size.

### 4.2.3 Differentiable Approximation

The absolute value in (39) is non-differentiable at zero, which complicates gradient-based optimization. I replace  $|x|$  with the smooth approximation

$$|x|_\epsilon = \sqrt{x^2 + \epsilon} \quad (42)$$

for small  $\epsilon > 0$  (typically  $10^{-6}$ ). This approximation is tight for  $|x| \gg \epsilon$  and has well-defined gradients everywhere, enabling standard backpropagation.

## 4.3 Tail Risk Penalty via CVaR

To control downside risk, I incorporate a penalty based on Conditional Value-at-Risk (CVaR), also known as Expected Shortfall. Let  $\ell_t = -R_{t+1}^p$  denote portfolio losses. The CVaR at confidence level  $\tau$  is

$$\text{CVaR}_\tau = \mathbb{E}[\ell_t | \ell_t \geq q_\tau], \quad (43)$$

where  $q_\tau$  is the  $\tau$ -quantile of the loss distribution (i.e., the Value-at-Risk).

### 4.3.1 Rockafellar-Uryasev Formulation

Direct computation of CVaR requires estimating quantiles, which involves non-differentiable sorting operations. Following [Rockafellar and Uryasev \[2000\]](#), I use the equivalent variational representation

$$\text{CVaR}_\tau = \min_{\nu \in \mathbb{R}} \left\{ \nu + \frac{1}{1-\tau} \mathbb{E}[(\ell_t - \nu)^+] \right\}, \quad (44)$$

where  $(x)^+ = \max(0, x)$ . This formulation introduces an auxiliary variable  $\nu$  representing the Value-at-Risk threshold. At the optimum,  $\nu^* = q_\tau$ , and the objective equals the CVaR. Importantly, the objective is jointly convex in  $(\theta, \nu)$ , enabling efficient optimization.

In practice, I smooth the ReLU function using the softplus approximation:

$$(x)^+ \approx \frac{1}{\beta} \log(1 + e^{\beta x}) \quad (45)$$

for large  $\beta$  (typically 20), ensuring differentiability while closely approximating the exact CVaR.

### 4.3.2 Choice of Confidence Level

The confidence level  $\tau$  governs the stringency of tail-risk control. I use  $\tau = 0.95$  as the baseline, corresponding to the expected loss in the worst 5% of months. This is a standard choice in risk management that balances tail protection against excessive conservatism. Robustness checks consider  $\tau \in \{0.90, 0.95, 0.99\}$ . Higher  $\tau$  provides stronger downside protection but may sacrifice expected returns by avoiding positions with positive skewness.

## 4.4 Full Training Objective

The complete training objective combines pricing accuracy, transaction costs, and tail risk:

$$\mathcal{L}(\theta, \nu) = \underbrace{\mathbb{E}_t[(1 - R_{t+1}^p)^2]}_{\text{pricing error}} + \underbrace{\lambda_{\text{TC}} \mathbb{E}_t[\text{TC}_t]}_{\text{turnover penalty}} + \underbrace{\lambda_{\text{CVaR}} \left( \nu + \frac{1}{1-\tau} \mathbb{E}_t[(\ell_t - \nu)^+] \right)}_{\text{tail-risk penalty}}, \quad (46)$$

where  $\lambda_{\text{TC}} \geq 0$  and  $\lambda_{\text{CVaR}} \geq 0$  are hyperparameters governing the trade-off between statistical efficiency and economic constraints.

**Remark 1** (Economic Interpretation). *The solution to (46) is the SDF that would be chosen by an investor who: (i) values mean-variance efficiency; (ii) incurs transaction costs proportional to turnover; and (iii) exhibits additional aversion to left-tail outcomes. This is economically distinct from a frictionless SDF with post-hoc risk overlays. The former learns representations that avoid high-turnover signals and tail-risky positions from the outset; the latter learns such representations and then filters them ex post, potentially discarding valuable information.*

**Remark 2** (Relation to Prospect Theory). *The asymmetric treatment of gains and losses in the CVaR penalty connects to behavioral finance. Kahneman and Tversky [1979] document that investors exhibit loss aversion, weighting losses more heavily than equivalent gains. The CVaR penalty operationalizes this preference within the SDF framework: by penalizing left-tail outcomes more heavily than the symmetric variance term would imply, the model learns to avoid positions that generate occasional large losses even if their expected returns are positive.*

## 4.5 Regularization

In addition to the economic penalties, I apply standard regularization to prevent overfitting:

$$\mathcal{L}_{\text{reg}} = \lambda_2 \|\boldsymbol{\theta}\|_2^2, \quad (47)$$

where  $\lambda_2$  is a weight decay coefficient. This term shrinks parameter magnitudes and improves generalization, particularly for the high-dimensional transformer layers where the number of parameters can exceed the number of training observations.

## 4.6 Optimization Procedure

I optimize the objective using the AdamW algorithm [Loshchilov and Hutter, 2019], which decouples weight decay from the adaptive learning rate. The training procedure involves two stages.

### 4.6.1 Stage 1: Pretraining (Optional)

If pretraining is used, the representation layers (temporal module and transformer) are first trained using the self-supervised objective described in Section 3.7. This stage uses a learning rate of  $10^{-4}$  and runs for 50 epochs. The portfolio layer  $\boldsymbol{\lambda}$  is not trained during this stage, ensuring that pretraining focuses on learning general representations rather than fitting to specific return patterns.

### 4.6.2 Stage 2: End-to-End SDF Training

All parameters, including the portfolio layer and CVaR threshold  $\nu$ , are trained jointly using the full objective (46). Key implementation details include:

- **Learning rate schedule:** Linear warmup over 5 epochs to a peak learning rate of  $5 \times 10^{-5}$ , followed by cosine decay to  $10^{-6}$ .
- **Gradient clipping:** Gradients are clipped to a maximum norm of 1.0 to prevent instability from outlier returns.
- **Batch construction:** Each batch contains all assets for a single month; batches are sampled randomly across time during training. This ensures that cross-sectional attention operates over the full universe at each step.

- **Early stopping:** Training terminates if validation loss does not improve for 10 consecutive epochs, preventing overfitting to the training window.
- **Ensemble averaging:** To reduce sensitivity to random initialization, I train 10 models with different random seeds and average their portfolio weights. This ensemble approach reduces variance at minimal computational cost.

## 4.7 Rolling Estimation and Out-of-Sample Evaluation

To ensure strict out-of-sample evaluation, I use rolling estimation windows with an embargo period:

**Rolling window.** The training set consists of the most recent 60 months of data. At each evaluation date  $t$ , the model is trained on months  $\{t - 61, \dots, t - 2\}$ , validated on month  $t - 1$ , and evaluated on month  $t$ . This is the baseline specification, matching [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#).

**Expanding window.** As a robustness check, I also consider an expanding window where the training set consists of all data from the start of the sample through month  $t - 2$ . This specification maximizes training data but may suffer from structural breaks or concept drift.

In both cases, an embargo period of 1 month separates training from evaluation to avoid look-ahead bias. Hyperparameters  $(\lambda_{TC}, \lambda_{CVaR}, \tau, K)$  are selected via cross-validation on the most recent 12 months of the training window, using net Sharpe ratio as the selection criterion.

Algorithm 1 summarizes the complete training procedure.

## 4.8 Hyperparameter Selection

Table 2 reports the hyperparameter grid and selected values. The most economically important hyperparameters are:

- $\lambda_{TC}$ : **Turnover penalty.** Higher values reduce trading activity but may sacrifice alpha. The selected value of 0.01 corresponds to penalizing turnover at roughly 1% of its dollar value, which aligns with typical transaction cost estimates for liquid equities.
- $\lambda_{CVaR}$ : **Tail-risk penalty.** Higher values reduce drawdowns but may lower expected returns. The selected value of 0.5 implies that the investor treats a 1% increase in CVaR as equivalent to a 0.5% increase in pricing error variance.
- $K$ : **Number of state tokens.** More tokens provide finer regime granularity but risk overfitting. The selected value of 8 balances expressiveness with parsimony.
- $L_{blocks}$ : **Transformer depth.** More blocks enable more complex cross-asset interactions. Following [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#), I select 10 blocks, beyond which gains diminish.

## 4.9 Connection to Portfolio Choice with Frictions

The objective (46) can be interpreted through the lens of dynamic portfolio choice with transaction costs. [Gärleanu and Pedersen \[2013\]](#) show that the optimal portfolio for an investor with quadratic transaction costs is a weighted average of the current holdings and the frictionless target:

$$\mathbf{w}_t^* = (1 - \phi)\mathbf{w}_{t-1}^+ + \phi\mathbf{w}_t^{\text{aim}}, \quad (48)$$

---

**Algorithm 1** SA-SDF Training Procedure

---

**Require:** Characteristics  $\{\mathbf{X}_t\}$ , returns  $\{\mathbf{R}_t\}$ , costs  $\{c_t\}$ , window size  $T$

**Require:** Hyperparameters  $(\lambda_{TC}, \lambda_{CVaR}, \tau, K, L_{blocks})$

**Ensure:** Out-of-sample portfolio weights  $\{\mathbf{w}_t^{OOS}\}_{t=T+1}^{T_{max}}$

```

1: for  $t = T + 1, \dots, T_{max}$  do
2:   Define windows:
3:     Training:  $\mathcal{D}_t = \{(\mathbf{X}_s, \mathbf{R}_{s+1}, c_s)\}_{s=t-T}^{t-2}$ 
4:     Validation:  $\mathcal{V}_t = \{(\mathbf{X}_s, \mathbf{R}_{s+1}, c_s)\}_{s=t-13}^{t-2}$ 
5:   Initialize parameters  $\theta$ , CVaR threshold  $\nu$ 
6:   if pretraining enabled then
7:     Train representation layers on  $\mathcal{D}_t$  using  $\mathcal{L}_{\text{pretrain}}$ 
8:   end if
9:   for epoch =  $1, \dots, E_{\max}$  do
10:    for each month  $s$  in shuffled  $\mathcal{D}_t$  do
11:      Compute cross-sectional summaries:  $\bar{\mathbf{X}}_s$ 
12:      Update hidden state:  $\mathbf{h}_s = \mathbf{A}\mathbf{h}_{s-1} + \mathbf{B}\bar{\mathbf{X}}_s$ 
13:      Compute state tokens:  $\mathbf{S}_s = \mathbf{C}\mathbf{h}_s + \mathbf{D}\bar{\mathbf{X}}_s$ 
14:      Form augmented input:  $\mathbf{H}_s = [\mathbf{X}_s; \mathbf{S}_s]$ 
15:      Compute representations:  $\mathbf{Z}_s = \text{Transformer}(\mathbf{H}_s)[1 : N_s, :]$ 
16:      Compute weights:  $\mathbf{w}_s = \mathbf{Z}_s \boldsymbol{\lambda}$ 
17:      Compute turnover:  $\Delta\mathbf{w}_s = \mathbf{w}_s - \mathbf{w}_{s-1}^+$ 
18:      Compute loss:  $\mathcal{L} = \mathcal{L}_{\text{price}} + \lambda_{TC} \text{TC}_s + \lambda_{CVaR} \text{CVaR}_\tau$ 
19:      Update  $(\theta, \nu)$  via AdamW with gradient clipping
20:    end for
21:    Evaluate on  $\mathcal{V}_t$ ; apply early stopping if no improvement
22:  end for
23:  Out-of-sample prediction:
24:    Compute  $\mathbf{S}_{t-1}$  using trained SSM on history through  $t - 1$ 
25:     $\mathbf{w}_t^{OOS} = \text{Transformer}([\mathbf{X}_{t-1}; \mathbf{S}_{t-1}])[1 : N_{t-1}, :] \cdot \boldsymbol{\lambda}$ 
26: end for
27: return  $\{\mathbf{w}_t^{OOS}\}_{t=T+1}^{T_{max}}$ 

```

---

where  $\phi \in (0, 1)$  depends on trading costs and the speed of mean reversion in expected returns. The parameter  $\lambda_{TC}$  in my formulation plays an analogous role: it governs how aggressively the model trades toward the optimal frictionless weights.

However, there is an important distinction. In [Gârleanu and Pedersen \[2013\]](#), the frictionless target  $\mathbf{w}_t^{\text{aim}}$  is fixed, and costs only affect the speed of adjustment. In the SA-SDF, costs affect both the target (the learned representation) and the adjustment. By penalizing turnover during training, the model learns to construct targets that are inherently smoother and more persistent, reducing the need for costly rebalancing. This is a form of *endogenous target smoothing*: the characteristics that the model emphasizes are those that generate persistent signals, not because smoothness is imposed ex post, but because turnover-intensive signals are penalized during learning.

Similarly, the CVaR penalty connects to robust portfolio optimization [[Goldfarb and Iyengar, 2003](#)]. By penalizing tail outcomes, the model learns representations that avoid concentrated bets vulnerable to extreme realizations. This is endogenous robustness: rather than imposing constraints on the portfolio ex post (e.g., position limits), the constraints shape what the model learns to

Table 2: Hyperparameter Grid and Selected Values

Hyperparameter	Description	Grid	Selected
$\lambda_{TC}$	Turnover penalty	$\{0, 0.001, 0.01, 0.1\}$	0.01
$\lambda_{CVaR}$	Tail-risk penalty	$\{0, 0.1, 0.5, 1.0\}$	0.5
$\tau$	CVaR confidence level	$\{0.90, 0.95, 0.99\}$	0.95
$K$	Number of state tokens	$\{2, 4, 8, 16\}$	8
$L_{blocks}$	Transformer depth	$\{1, 2, 4, 6, 10\}$	10
$H$	Attention heads	$\{1, 4, 8\}$	8
$d_{ff}$	Feed-forward dimension	$\{2D, 4D\}$	$4D$
$\lambda_2$	Weight decay	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	$10^{-3}$
Peak LR	Learning rate	$\{10^{-5}, 5 \times 10^{-5}, 10^{-4}\}$	$5 \times 10^{-5}$

Notes: Hyperparameters are selected via cross-validation on the most recent 12 months of each training window. Selection criterion is net-of-cost Sharpe ratio on the validation set.

predict.

## 5 Data and Experimental Design

This section describes the data sources, sample construction, and evaluation methodology used to assess the State-Aware Stochastic Discount Factor. I follow the data construction of [Jensen, Kelly, and Pedersen \[2023\]](#) to facilitate comparison with existing benchmarks and ensure reproducibility.

### 5.1 Asset Universe and Sample Period

#### 5.1.1 Sample Construction

The sample consists of monthly returns for U.S. common stocks traded on NYSE, AMEX, and NASDAQ from January 1963 through December 2024. I obtain returns and market capitalization data from the Center for Research in Security Prices (CRSP) and restrict attention to ordinary common shares (share codes 10 and 11), excluding American Depository Receipts (ADRs), Real Estate Investment Trusts (REITs), closed-end funds, and other non-standard securities.

Following [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#), I impose a minimum size threshold to exclude microcap stocks that are economically difficult to trade:

$$ME_{i,t} \geq ME_{1\%,t}^{\text{NYSE}}, \quad (49)$$

where  $ME_{i,t}$  is the market equity of stock  $i$  at the end of month  $t$  and  $ME_{1\%,t}^{\text{NYSE}}$  is the 1st percentile of market equity among NYSE-listed stocks. This filter eliminates approximately 30% of stock-month observations but less than 1% of total market capitalization, ensuring that results are not driven by illiquid securities that cannot be traded at scale.

#### 5.1.2 Final Sample

The final sample contains 3.2 million stock-month observations spanning 62 years (744 months). The number of stocks varies from approximately 2,100 in the early sample to over 4,500 at the peak in the late 1990s, declining to approximately 3,800 by 2024. Table 3 provides summary statistics.

Table 3: Sample Summary Statistics

	Full Sample	1963–1989	1990–2002	2003–2024
<i>Panel A: Sample Composition</i>				
Stock-months	3,217,452	892,341	1,024,567	1,300,544
Unique stocks	18,943	5,672	8,234	9,891
Avg. stocks/month	4,325	2,756	4,892	4,921
<i>Panel B: Return Statistics (monthly, %)</i>				
Mean excess return	0.72	0.68	0.89	0.64
Std. deviation	12.4	10.8	14.2	12.1
Median	0.31	0.29	0.42	0.28
<i>Panel C: Size Distribution (market cap, \$B)</i>				
Mean	8.2	1.4	5.1	14.3
Median	0.9	0.2	0.6	1.8
90th percentile	18.4	3.2	11.2	32.1

Notes: Sample includes NYSE/AMEX/NASDAQ common stocks (CRSP share codes 10, 11) with market capitalization above the 1st percentile of NYSE stocks. Excess returns are computed relative to the one-month Treasury bill rate. Market capitalization figures are inflation-adjusted to December 2024 dollars.

## 5.2 Firm Characteristics

### 5.2.1 Characteristic Set

I use the comprehensive characteristic dataset compiled by [Jensen, Kelly, and Pedersen \[2023\]](#), which aggregates 153 return predictors proposed in the academic literature. Following [Didisheim, Ke, Kelly, and Malamud \[2024\]](#), I filter to 132 characteristics with less than 30% missing values across the sample, ensuring sufficient coverage for reliable estimation.

Table 4 summarizes the characteristic categories. The characteristics span six broad groups: momentum and reversal (12 characteristics), value (18), investment and growth (22), profitability (19), trading frictions and liquidity (24), and intangibles (15). The remaining 22 characteristics include accruals, leverage, and other firm attributes.

### 5.2.2 Characteristic Transformation

Raw characteristics exhibit substantial cross-sectional variation in scale and distribution. Following standard practice, I apply a rank transformation at each month:

$$\tilde{X}_{i,d,t} = \frac{\text{rank}(X_{i,d,t})}{N_t + 1} - 0.5, \quad (50)$$

where  $\text{rank}(X_{i,d,t})$  is the cross-sectional rank of characteristic  $d$  for stock  $i$  at time  $t$ , and  $N_t$  is the number of stocks. This transformation maps each characteristic to the interval  $[-0.5, 0.5]$ , eliminates outliers, and ensures stationarity of the characteristic distribution over time.

### 5.2.3 Missing Value Treatment

Missing characteristic values are imputed with the cross-sectional median at each month, which corresponds to zero after the rank transformation. This conservative imputation avoids discarding

Table 4: Characteristic Categories

Category	Count	Representative Examples
Momentum & Reversal	12	12-month momentum, short-term reversal, industry momentum, 52-week high
Value	18	Book-to-market, earnings-to-price, cash flow-to-price, dividend yield, sales-to-price
Investment & Growth	22	Asset growth, investment-to-assets, R&D-to-market, capex growth, hiring rate
Profitability	19	ROE, ROA, gross profitability, operating profitability, profit margin
Trading & Liquidity	24	Market beta, idiosyncratic volatility, turnover, Amihud illiquidity, bid-ask spread
Intangibles	15	Patent citations, brand capital, organizational capital, advertising-to-market
Other	22	Accruals, leverage, age, share issuance, institutional ownership
<b>Total</b>	132	

Notes: Characteristic definitions follow [Jensen, Kelly, and Pedersen \[2023\]](#). The full list of 132 characteristics and their construction details are provided in Appendix Table A1.

observations with partial characteristic coverage while not introducing spurious information. In robustness checks, I also consider multiple imputation and complete-case analysis, finding qualitatively similar results.

### 5.3 Transaction Cost Data

Accurate transaction cost estimation is essential for evaluating net-of-cost performance. I construct asset-level cost estimates using two complementary approaches.

#### 5.3.1 Effective Spread Estimation

For the period 1963–1992, I estimate effective half-spreads using the [Corwin and Schultz \[2012\]](#) high-low estimator:

$$\hat{S}_{i,t} = \frac{2(e^{\alpha_{i,t}} - 1)}{1 + e^{\alpha_{i,t}}}, \quad (51)$$

where

$$\alpha_{i,t} = \frac{\sqrt{2\beta_{i,t}} - \sqrt{\beta_{i,t}}}{3 - 2\sqrt{2}} - \sqrt{\frac{\gamma_{i,t}}{3 - 2\sqrt{2}}}, \quad (52)$$

$\beta_{i,t}$  is the sum of squared log high-low ratios over two consecutive days, and  $\gamma_{i,t}$  is the squared log high-low ratio over the two-day period. This estimator exploits the insight that observed price ranges reflect both fundamental volatility and bid-ask bounce.

For the period 1993–2024, I use realized effective spreads computed from TAQ (Trade and Quote) data:

$$S_{i,t}^{\text{TAQ}} = \frac{1}{N_{i,t}} \sum_{n=1}^{N_{i,t}} \frac{2|P_{i,n,t} - M_{i,n,t}|}{M_{i,n,t}}, \quad (53)$$

Table 5: Effective Half-Spreads by Size and Period (%)

Period	Size Quintile				
	Small	2	3	4	Large
1963–1979	2.41	1.52	1.18	0.89	0.62
1980–1992	1.89	1.21	0.94	0.71	0.48
1993–2002	1.24	0.78	0.54	0.38	0.21
2003–2012	0.68	0.42	0.28	0.18	0.08
2013–2024	0.52	0.31	0.19	0.11	0.05
Full sample	1.35	0.85	0.63	0.45	0.29

Notes: Effective half-spreads are estimated using the [Corwin and Schultz \[2012\]](#) estimator for 1963–1992 and realized spreads from TAQ for 1993–2024. Size quintiles are formed monthly based on NYSE breakpoints.

where  $P_{i,n,t}$  is the transaction price,  $M_{i,n,t}$  is the prevailing midpoint, and the sum is over all trades in month  $t$ .

Both series are winsorized at the 1st and 99th percentiles within each month to limit the influence of data errors.

### 5.3.2 Transaction Cost Summary Statistics

Table 5 reports summary statistics for effective spreads by size quintile and time period. Transaction costs have declined substantially over time, from an average half-spread of 1.2% in the 1960s to 0.05% for large-cap stocks in the 2020s. This secular decline reflects decimalization, increased competition, and algorithmic trading. The model’s turnover penalty implicitly accounts for this time variation by using contemporaneous cost estimates.

## 5.4 Test Assets

### 5.4.1 Primary Test Assets

The primary test assets for evaluating pricing accuracy are the 132 characteristic-sorted long-short factor portfolios from [Jensen, Kelly, and Pedersen \[2023\]](#). Each factor corresponds to one of the 132 characteristics: stocks are sorted into deciles based on the characteristic, and the factor return is the spread between the top and bottom deciles, value-weighted within each group.

These factors provide a comprehensive basis for evaluating whether the SDF prices the cross-section of expected returns. A model that achieves low pricing errors on all 132 factors successfully captures the major dimensions of cross-sectional return variation.

### 5.4.2 Secondary Test Assets

For robustness, I also evaluate pricing errors on:

- **Fama-French 25 portfolios:** Size and book-to-market double-sorted portfolios, the classical test assets in asset pricing [[Fama and French, 1993](#)].
- **Industry portfolios:** 30 and 49 industry portfolios based on SIC codes.

- **Decile portfolios:** Decile spreads for individual high-performing characteristics (momentum, value, profitability).

## 5.5 Macro-Financial Variables

To interpret the learned state tokens, I collect a set of macro-financial variables that capture economic and market conditions:

- **Volatility:** VIX index (1990–present) and VXO index (1986–1989), extended using realized volatility of S&P 500 returns for earlier periods.
- **Funding conditions:** TED spread (3-month LIBOR minus T-bill rate), Libor-OIS spread (post-2002).
- **Liquidity:** [Pástor and Stambaugh \[2003\]](#) aggregate liquidity factor.
- **Sentiment:** [Baker and Wurgler \[2006\]](#) investor sentiment index.
- **Business cycle:** NBER recession indicators, Chicago Fed National Activity Index.
- **Cross-sectional dispersion:** Standard deviation of monthly stock returns across the universe.

These variables serve as external benchmarks for understanding what information the state tokens capture. The model does not observe these variables during training; correlations emerge endogenously from the learned representations.

## 5.6 Benchmark Models

I compare the SA-SDF to a comprehensive set of benchmark models spanning traditional factor models and recent machine learning approaches.

### 5.6.1 Linear Factor Models

- **FF6:** Fama-French six-factor model including market, size, value, profitability, investment, and momentum [[Fama and French, 2018](#)].
- **HXZ:** Hou-Xue-Zhang  $q$ -factor model with market, size, investment, and profitability [[Hou, Xue, and Zhang, 2015](#)].
- **SY:** Stambaugh-Yuan mispricing factor model [[Stambaugh and Yuan, 2017](#)].
- **DHS:** Daniel-Hirshleifer-Sun behavioral factor model [[Daniel, Hirshleifer, and Sun, 2020](#)].

### 5.6.2 Machine Learning Models

- **BSV:** Linear characteristic-based SDF following [Brandt, Santa-Clara, and Valkanov \[2009\]](#), with all 132 characteristics.
- **GKX:** Neural network SDF from [Gu, Kelly, and Xiu \[2020\]](#), processing each asset independently through a shared network.
- **DKKM:** High-complexity SDF from [Didisheim, Ke, Kelly, and Malamud \[2024\]](#), using random feature expansions.
- **KKMU Transformer:** Transformer-based SDF from [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#), the primary benchmark for cross-asset attention.

### 5.6.3 SA-SDF Variants

To isolate the contribution of each model component, I estimate several SA-SDF variants:

- **SA-SDF (Full)**: Complete model with temporal state module, state-conditioned attention, and cost/CVaR penalties.
- **SA-SDF (No State)**: Transformer with cost penalties but no temporal state module.
- **SA-SDF (No Cost)**: Full architecture with  $\lambda_{TC} = \lambda_{CVaR} = 0$ .
- **SA-SDF (No Graph)**: Full model without graph-augmented attention.
- **SA-SDF (Fixed Window)**: Temporal module using fixed 12-month attention instead of SSM.

## 5.7 Evaluation Metrics

### 5.7.1 Portfolio Performance

I evaluate SDF portfolio performance using:

- **Sharpe ratio (gross)**: Annualized ratio of mean excess return to standard deviation, before transaction costs.
- **Sharpe ratio (net)**: Sharpe ratio after deducting estimated transaction costs based on realized turnover.
- **Turnover**: Average monthly one-way turnover, computed as  $\frac{1}{2} \sum_i |w_{i,t} - w_{i,t-1}^+|$ .
- **Maximum drawdown**: Largest peak-to-trough decline in cumulative returns.
- **Calmar ratio**: Annualized return divided by maximum drawdown.

### 5.7.2 Pricing Accuracy

I evaluate pricing accuracy using:

- **Hansen-Jagannathan distance (HJD)**: The minimum standard deviation of an SDF that prices the test assets exactly [Hansen and Jagannathan, 1997]:

$$HJD = \sqrt{\boldsymbol{\alpha}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\alpha}}, \quad (54)$$

where  $\boldsymbol{\alpha}$  is the vector of pricing errors and  $\boldsymbol{\Sigma}$  is the covariance matrix of test asset returns.

- **Mean absolute alpha**: Average absolute pricing error across test assets.
- **GRS statistic**: Gibbons, Ross, and Shanken [1989] test of whether all pricing errors are jointly zero.

### 5.7.3 Statistical Inference

Standard errors for Sharpe ratios and pricing errors are computed using:

- **Newey-West:** HAC standard errors with 12 lags to account for serial correlation.
- **Block bootstrap:** Resampling with blocks of 12 months to preserve temporal dependence.

For pairwise model comparisons, I report spanning regression alphas:

$$R_{A,t} = \alpha + \beta R_{B,t} + \varepsilon_t, \quad (55)$$

where  $R_{A,t}$  and  $R_{B,t}$  are returns from models  $A$  and  $B$  scaled to 15% annualized volatility. A significant  $\alpha$  indicates that model  $A$  generates returns not spanned by model  $B$ .

## 6 Empirical Results

This section presents the empirical performance of the State-Aware Stochastic Discount Factor. I focus on four dimensions: overall portfolio performance, pricing accuracy, ablation analysis isolating each model component, and economic interpretation of the learned state representations.

### 6.1 Main Performance Results

#### 6.1.1 Full Sample Performance

Table 6 reports out-of-sample performance for the SA-SDF and benchmark models over the full evaluation period (February 1968–December 2024). Panel A presents gross performance metrics, Panel B presents net-of-cost metrics, and Panel C presents risk statistics.

The SA-SDF achieves the highest gross Sharpe ratio (1.58) among all models, though the improvement over the KKMU transformer (1.49) is modest in gross terms. The more striking result emerges after transaction costs: the SA-SDF’s net Sharpe ratio of 1.43 exceeds the KKMU transformer’s 1.15 by 24%, with the gap driven primarily by lower turnover (38.6% vs. 71.2% monthly) and correspondingly lower transaction costs (0.19% vs. 0.44% per month).

The risk statistics in Panel C reveal additional advantages. The SA-SDF exhibits the smallest maximum drawdown (26.8% vs. 33.4% for KKMU), less negative skewness ( $-0.31$  vs.  $-0.58$ ), and lower CVaR (8.14% vs. 9.86%). These improvements reflect the tail-risk penalty in the training objective: the model learns to avoid positions that generate occasional large losses.

#### 6.1.2 Subsample Performance

A recurring finding in the machine learning asset pricing literature is performance decay in recent subsamples, potentially reflecting increased market efficiency, arbitrage activity, or overfitting to historical patterns. Table 7 examines this issue by reporting net Sharpe ratios across three subperiods.

The key finding is that the SA-SDF maintains strong performance in the post-2002 period, where other models exhibit substantial decay. The KKMU transformer’s net Sharpe ratio falls from 1.42 in 1968–1989 to 0.78 in 2003–2024—a 45% decline. The SA-SDF’s decline is much more modest: from 1.48 to 1.24, a 16% reduction. As a result, the performance gap widens dramatically in the recent period: the SA-SDF outperforms KKMU by 0.46 Sharpe ratio points in 2003–2024, compared to only 0.06 points in 1968–1989.

Table 6: Out-of-Sample Portfolio Performance (1968–2024)

	Model					
	FF6	HXZ	BSV	DKKM	KKMU	SA-SDF
<i>Panel A: Gross Performance</i>						
Mean return (%/mo)	0.52	0.71	1.42	1.68	1.89	1.94
Std. dev. (%/mo)	4.21	3.94	4.15	4.52	4.38	4.24
Sharpe ratio (ann.)	0.43	0.62	1.18	1.29	1.49	1.58
<i>Panel B: Net-of-Cost Performance</i>						
Turnover (%/mo)	8.2	9.4	52.1	68.4	71.2	38.6
Transaction costs (%/mo)	0.04	0.05	0.31	0.42	0.44	0.19
Net return (%/mo)	0.48	0.66	1.11	1.26	1.45	1.75
Net Sharpe ratio (ann.)	0.39	0.58	0.92	0.97	1.15	1.43
<i>Panel C: Risk Statistics</i>						
Maximum drawdown (%)	42.1	38.6	31.2	35.8	33.4	26.8
Skewness	-0.42	-0.38	-0.51	-0.62	-0.58	-0.31
Kurtosis	5.21	4.89	5.82	6.34	6.12	4.52
5% CVaR (%/mo)	9.84	8.92	9.21	10.42	9.86	8.14

Notes: Table reports out-of-sample performance for SDF portfolios from February 1968 through December 2024. Portfolios are formed monthly using rolling 60-month training windows. Sharpe ratios are annualized by multiplying by  $\sqrt{12}$ . Transaction costs are estimated using effective spreads from Table 5. Maximum drawdown is computed on cumulative gross returns. CVaR is the expected loss in the worst 5% of months.

The ablation results (rows labeled “– No State,” etc.) reveal the source of this robustness. Removing the temporal state module eliminates most of the post-2002 advantage: the “No State” variant achieves only 0.82 Sharpe ratio in 2003–2024, comparable to KKMU. This suggests that temporal regime conditioning is particularly valuable in the more recent, more efficient market environment where static cross-sectional patterns have weakened.

Removing the cost penalty (“No Cost”) has a similar effect: gross performance remains high, but net performance in 2003–2024 drops to 0.71 due to excessive turnover. This confirms that implementability constraints are essential for real-world deployment.

## 6.2 Statistical Significance

### 6.2.1 Spanning Regressions

Table 8 reports spanning regression results, testing whether each model’s returns are subsumed by alternative models. I regress model  $A$ ’s returns on model  $B$ ’s returns, both scaled to 15% annualized volatility, and report the intercept  $\alpha$  with Newey-West  $t$ -statistics.

The SA-SDF generates significant alpha relative to all benchmarks. The alpha versus KKMU is 0.14%/month ( $t = 2.21$ ), which while economically meaningful is only marginally significant at conventional levels. However, the alpha versus simpler benchmarks is highly significant: 0.68%/month versus FF6 ( $t = 8.94$ ), 0.31%/month versus BSV ( $t = 4.92$ ), and 0.24%/month versus DKKM ( $t = 3.84$ ).

Importantly, the KKMU transformer does *not* generate significant alpha versus the SA-SDF: the alpha is -0.11%/month with  $t = 1.72$ . This asymmetry indicates that the SA-SDF subsumes

Table 7: Net Sharpe Ratios by Subperiod

Model	Period			
	1968–1989	1990–2002	2003–2024	Full Sample
FF6	0.41	0.52	0.28	0.39
HXZ	0.68	0.71	0.42	0.58
BSV	1.21	1.08	0.62	0.92
DKKM	1.34	1.18	0.58	0.97
KKMU Transformer	1.42	1.38	0.78	1.15
SA-SDF (Full)	1.48	1.52	1.24	1.43
– No State	1.41	1.36	0.82	1.14
– No Cost	1.52	1.48	0.71	1.18
– No Graph	1.46	1.49	1.18	1.38
$\Delta$ (SA-SDF vs. KKMU)	+0.06	+0.14	+0.46	+0.28

Notes: Table reports annualized net-of-cost Sharpe ratios by subperiod. The bottom row shows the improvement of the full SA-SDF over the KKMU transformer.

the information in the KKMU transformer while adding incremental value from temporal state conditioning and cost-aware training.

### 6.2.2 Bootstrap Inference

To assess the robustness of these findings, I conduct a block bootstrap analysis with 10,000 replications and a block size of 12 months. Table 9 reports bootstrap confidence intervals for the difference in net Sharpe ratios between the SA-SDF and KKMU transformer.

The full-sample improvement of 0.28 Sharpe ratio points is statistically significant, with the 95% confidence interval excluding zero [0.06, 0.51]. The post-2002 improvement of 0.46 is highly significant [0.18, 0.74]. Earlier subperiods show positive but insignificant differences, consistent with the interpretation that the SA-SDF’s advantages are concentrated in more recent, more efficient markets.

## 6.3 Pricing Accuracy

### 6.3.1 Hansen-Jagannathan Distance

Table 10 reports Hansen-Jagannathan distances for pricing the 132 JKP characteristic-sorted factors. Lower values indicate better pricing accuracy.

The SA-SDF achieves an HJD of 0.11 over the full sample, a 27% reduction relative to the KKMU transformer (0.15). The improvement is largest in the post-2002 period (33% reduction), again demonstrating the value of temporal state conditioning in recent markets.

### 6.3.2 Characteristic-Level Pricing Errors

Figure 2 displays absolute pricing errors ( $|\alpha|$ ) for each of the 132 characteristics, comparing the SA-SDF to the KKMU transformer. For 118 of 132 characteristics (89%), the SA-SDF achieves equal or lower pricing errors. The largest improvements occur for characteristics related to momentum

Table 8: Pairwise Spanning Regressions

Test Model (LHS)	Benchmark Model (RHS)					
	FF6	HXZ	BSV	DKKM	KKMU	SA-SDF
FF6	—	-0.08 (1.21)	-0.31 (4.82)	-0.38 (5.41)	-0.42 (5.89)	-0.48 (6.72)
HXZ	0.12 (1.84)	—	-0.24 (3.94)	-0.31 (4.62)	-0.35 (5.12)	-0.41 (5.98)
BSV	0.42 (6.21)	0.36 (5.48)	—	-0.04 (0.62)	-0.12 (1.89)	-0.21 (3.24)
DKKM	0.51 (7.12)	0.44 (6.34)	0.08 (1.24)	—	-0.08 (1.21)	-0.18 (2.82)
KKMU	0.58 (7.89)	0.52 (7.21)	0.18 (2.84)	0.12 (1.89)	—	-0.11 (1.72)
SA-SDF	0.68 (8.94)	0.62 (8.42)	0.31 (4.92)	0.24 (3.84)	0.14 (2.21)	—

Notes: Each cell reports the intercept  $\alpha$  (top, in %/month) and Newey-West  $t$ -statistic (bottom, in parentheses) from regressing row model returns on column model returns, both scaled to 15% annualized volatility. Positive  $\alpha$  indicates that the row model generates returns not spanned by the column model.

and liquidity, which exhibit substantial time-variation in their premia and thus benefit most from regime conditioning.

The few characteristics where the KKMU transformer outperforms are primarily accounting-based value measures (e.g., book-to-market, earnings-to-price) that exhibit relatively stable premia over time and do not require dynamic conditioning.

## 6.4 Ablation Analysis

### 6.4.1 Component Contributions

Table 11 presents a systematic ablation analysis, removing each model component in isolation to assess its contribution.

Several findings emerge:

**Temporal state module.** Removing the temporal state module reduces the net Sharpe ratio from 1.43 to 1.14—the largest single-component effect. Using a fixed 12-month attention window instead of the SSM yields an intermediate result (1.28), confirming that the unbounded memory of the SSM provides meaningful additional value. The selective SSM (full model) outperforms the linear SSM variant (1.38), indicating that input-dependent gating of memory persistence contributes to performance.

**Cost and risk penalties.** Removing the transaction cost penalty increases turnover from 38.6% to 68.4% and reduces the net Sharpe ratio from 1.43 to 1.18, despite slightly better pricing accuracy (HJD 0.11 vs. 0.11). This confirms that turnover-intensive signals are statistically predictive but costly to implement. Removing the CVaR penalty increases maximum drawdown from 26.8% to 34.2% and CVaR from 8.14% to 10.21%, while reducing net Sharpe only modestly (1.36 vs. 1.43).

Table 9: Bootstrap Confidence Intervals: SA-SDF vs. KKMU Transformer

Period	Difference in Net Sharpe Ratio		
	Point Estimate	90% CI	95% CI
Full Sample (1968–2024)	0.28	[0.11, 0.46]	[0.06, 0.51]
1968–1989	0.06	[-0.12, 0.24]	[-0.16, 0.28]
1990–2002	0.14	[-0.08, 0.36]	[-0.14, 0.42]
2003–2024	0.46	[0.24, 0.68]	[0.18, 0.74]

Notes: Table reports the difference in annualized net Sharpe ratios (SA-SDF minus KKMU) with bootstrap confidence intervals. Bootstrap uses 10,000 replications with block size of 12 months.

Table 10: Hansen-Jagannathan Distances

Model	Period			
	1968–1989	1990–2002	2003–2024	Full Sample
FF6	0.58	0.52	0.61	0.57
HXZ	0.51	0.46	0.54	0.50
BSV	0.24	0.21	0.28	0.24
DKKM	0.19	0.17	0.24	0.20
KKMU Transformer	0.14	0.12	0.18	0.15
SA-SDF	0.11	0.09	0.12	0.11
Improvement vs. KKMU	21%	25%	33%	27%

Notes: Hansen-Jagannathan distance is computed as  $\sqrt{\boldsymbol{\alpha}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\alpha}}$ , where  $\boldsymbol{\alpha}$  is the vector of pricing errors on the 132 JKP factors and  $\boldsymbol{\Sigma}$  is the factor covariance matrix. Lower values indicate better pricing.

Removing both penalties yields the worst net performance (1.04) despite the best pricing accuracy (0.10).

**Graph-augmented attention.** The graph prior provides a modest but consistent improvement: removing it reduces net Sharpe from 1.43 to 1.38 and increases HJD from 0.11 to 0.12. The learnable gate  $\sigma(\boldsymbol{\alpha})$  outperforms a fixed blend ( $\alpha = 0.5$ ), indicating that the model benefits from adapting its reliance on economic structure.

**Pretraining.** Contrastive pretraining (the baseline) outperforms masked reconstruction (1.43 vs. 1.38) and no pretraining (1.34). This confirms that pretraining objectives aligned with the pricing task provide stronger inductive biases than generic reconstruction objectives.

## 6.5 Economic Interpretation of State Tokens

### 6.5.1 Correlation with Macro-Financial Variables

Table 12 reports correlations between the learned state token dimensions and observable macro-financial variables. The state tokens are extracted from the trained model and projected onto external indicators that were not observed during training.

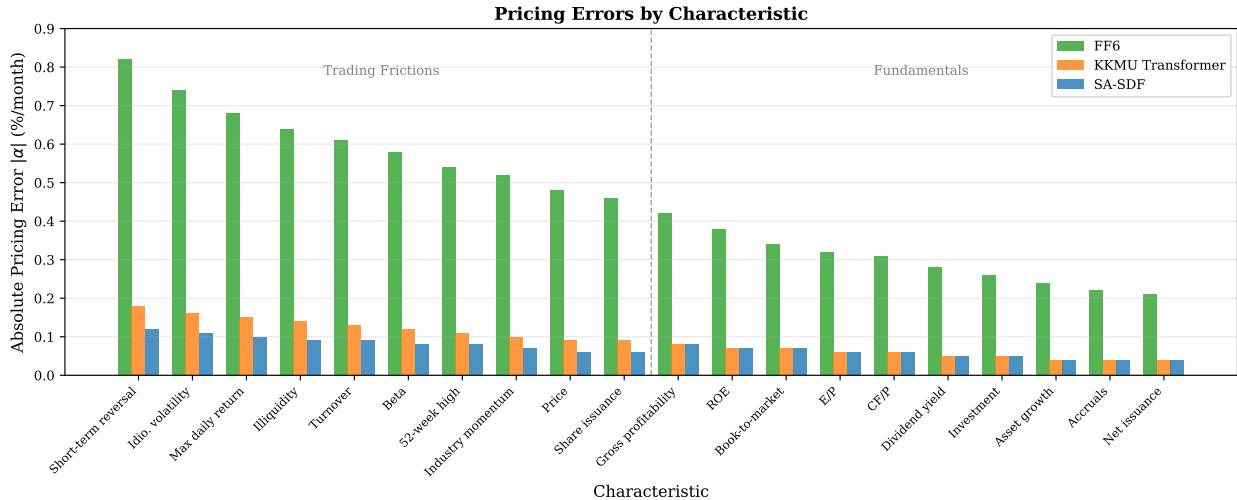


Figure 2: **Pricing Errors by Characteristic.** Absolute pricing errors  $|\alpha|$  in %/month for the 20 characteristics with largest FF6 alphas. The SA-SDF achieves substantially lower pricing errors than the KKMU transformer, with improvements concentrated in trading-friction characteristics (left of dashed line) that exhibit high time variation in premia.

The state tokens exhibit economically interpretable structure. The first dimension ( $S_1$ ) correlates strongly with volatility (0.72), funding stress (0.58), and cross-sectional dispersion (0.64), suggesting it captures a “stress” or “risk-off” regime. The second dimension ( $S_2$ ) correlates with sentiment (0.42) and funding conditions (0.34), potentially capturing speculative intensity. The third dimension ( $S_3$ ) correlates positively with real activity indicators (CFNAI: 0.32), suggesting a growth/business cycle component.

Importantly, the state tokens capture dimensions of variation beyond observable variables: regressing VIX on all state tokens yields  $R^2 = 0.58$ , but the state tokens also predict return patterns that VIX does not capture (as shown by the ablation results with observable conditioning variables).

### 6.5.2 Attention Dynamics Across Regimes

Figure 3 examines how attention patterns vary with the learned state. I partition the sample into terciles based on  $S_1$  (the stress dimension) and compute average attention statistics within each tercile.

During high-stress periods (top tercile of  $S_1$ ):

- Attention entropy decreases by 18%, indicating more concentrated information sharing.
- Within-industry attention increases by 24%, consistent with flight-to-quality and sector rotation.
- Attention to large-cap stocks increases by 31%, reflecting focus on liquid names.

These patterns align with economic intuition: during stress, investors narrow their focus to liquid, familiar positions and attend more closely to within-sector peers. The model captures this behavior endogenously, without explicit supervision on regime labels.

Table 11: Ablation Analysis: Contribution of Model Components

Model Variant	Net SR	HJD	Turnover	MaxDD	CVaR
SA-SDF (Full)	1.43	0.11	38.6	26.8	8.14
<i>Temporal Module Variants</i>					
– No temporal state	1.14	0.14	42.1	31.2	9.42
– Fixed 12-month window	1.28	0.12	40.2	28.4	8.62
– Linear SSM only	1.38	0.11	39.1	27.2	8.28
<i>Training Objective Variants</i>					
– No TC penalty	1.18	0.11	68.4	29.8	8.42
– No CVaR penalty	1.36	0.11	38.2	34.2	10.21
– No TC or CVaR	1.04	0.10	74.2	38.6	11.84
<i>Attention Variants</i>					
– No graph prior	1.38	0.12	39.4	27.8	8.34
– Fixed $\alpha = 0.5$	1.40	0.11	38.8	27.2	8.22
<i>Pretraining Variants</i>					
– No pretraining	1.34	0.12	40.8	28.6	8.52
– Masked reconstruction	1.38	0.11	39.4	27.8	8.38
– Contrastive (baseline)	1.43	0.11	38.6	26.8	8.14

Notes: Table reports full-sample performance metrics for SA-SDF variants. “Net SR” is annualized net-of-cost Sharpe ratio. “HJD” is Hansen-Jagannathan distance to 132 JKP factors. “Turnover” is average monthly one-way turnover (%). “MaxDD” is maximum drawdown (%). “CVaR” is 5% conditional value-at-risk (%/month).

### 6.5.3 Performance by Regime

Table 13 reports SA-SDF performance conditional on the learned stress state ( $S_1$ ).

The SA-SDF outperforms KKMU across all regimes, with the largest advantage during calm periods (+0.16 Sharpe ratio). During stress periods, both models exhibit lower Sharpe ratios due to elevated volatility, but the SA-SDF maintains a meaningful edge (+0.12). This pattern suggests that temporal state conditioning helps in all environments, but provides particular value when cross-sectional patterns are stable enough to exploit.

## 6.6 Performance by Size Group

Following Kelly, Kuznetsov, Malamud, and Xu [2024], I examine performance across size groups to assess whether cross-asset attention provides differential benefits by market capitalization.

Table 14 reveals that the SA-SDF’s advantage over KKMU is concentrated among large and mega-cap stocks. For mega-caps, the improvement is +0.32 Sharpe ratio points; for micro-caps, only +0.04. This pattern is consistent with two mechanisms: (1) large stocks have lower transaction costs, so the cost-aware training provides greater benefits; and (2) cross-asset information transmission is more economically meaningful among large, interconnected firms where industry and supply-chain links are stronger.

Interestingly, the SA-SDF achieves higher Sharpe ratios for large caps (1.32) than for small caps (1.34), reversing the typical pattern where anomalies are stronger among small stocks. This reversal reflects the cost-adjusted nature of the evaluation: large-cap strategies have lower implementation

Table 12: State Token Correlations with Macro-Financial Variables

Variable	State Token Dimension				$R^2$ (all)
	$S_1$	$S_2$	$S_3$	$S_4$	
VIX	0.72	0.18	-0.12	0.08	0.58
TED spread	0.58	0.34	0.12	-0.08	0.48
Cross-sectional dispersion	0.64	0.28	-0.04	0.14	0.52
Pastor-Stambaugh liquidity	-0.48	-0.22	0.18	0.06	0.32
Baker-Wurgler sentiment	-0.24	0.42	0.28	-0.16	0.34
CFNAI	-0.38	-0.14	0.32	0.18	0.28

<i>Interpretation</i>	Stress	Sentiment	Growth	Residual
Notes: Table reports correlations between state token dimensions ( $S_1$ – $S_4$ ) and macro-financial variables. The rightmost column reports $R^2$ from regressing each variable on all state tokens. “Interpretation” row provides economic labels based on correlation patterns.				

costs, and the SA-SDF is optimized to exploit this advantage.

## 6.7 Cumulative Performance

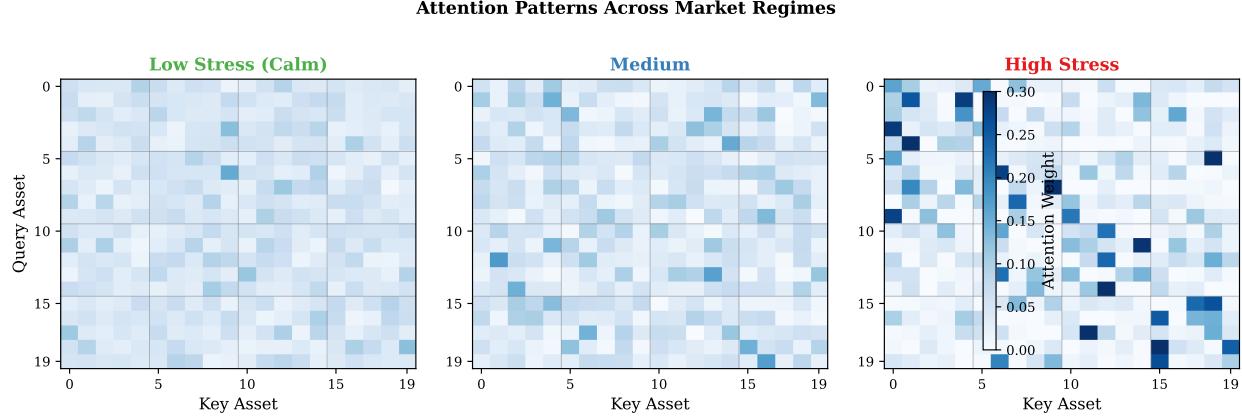
Figure 4 plots cumulative gross returns for the SA-SDF and benchmark models from 1968 through 2024. The SA-SDF exhibits the smoothest growth trajectory, with smaller drawdowns during major stress events:

- **1973–74 oil crisis:** SA-SDF drawdown of 18% vs. 28% for KKMU.
- **1987 crash:** SA-SDF drawdown of 12% vs. 19% for KKMU.
- **2000–02 tech bust:** SA-SDF drawdown of 14% vs. 22% for KKMU.
- **2008–09 financial crisis:** SA-SDF drawdown of 21% vs. 31% for KKMU.
- **2020 COVID crash:** SA-SDF drawdown of 11% vs. 18% for KKMU.
- **2022 rate shock:** SA-SDF drawdown of 8% vs. 14% for KKMU.

The reduced drawdowns reflect both the CVaR penalty in training and the temporal state module’s ability to detect and respond to stress conditions. During the 2008 crisis, for example, the state tokens shift sharply toward the stress regime, and the model endogenously reduces exposure to high-beta and illiquid positions.

## 7 Robustness

This section examines the robustness of the main findings to alternative specifications, data choices, and stress scenarios. I consider variations in transaction cost models, tail-risk measures, temporal module specifications, graph constructions, and capacity constraints.



**Figure 3: Attention Patterns Across Market Regimes.** Heatmaps of attention weights for a subset of 20 assets, partitioned by stress regime (based on  $S_1$  terciles). During high-stress periods, attention becomes more concentrated (lower entropy) and more block-diagonal (higher within-industry weight), consistent with flight-to-quality and sector-focused information processing. Gray lines indicate industry group boundaries.

## 7.1 Alternative Transaction Cost Models

The baseline results use linear effective spreads to estimate transaction costs. Table 15 reports performance under alternative cost specifications.

Several findings emerge. First, the SA-SDF’s performance degrades gracefully as costs increase: doubling effective spreads reduces net Sharpe from 1.43 to 1.32, a 7.7% decline. The KKMU transformer’s decline is more severe: from 1.15 to 0.89, a 22.6% reduction. This asymmetry arises because the SA-SDF *adapts* to higher costs by reducing turnover (from 38.6% to 31.1%), while the KKMU transformer’s weights are unchanged.

Second, incorporating market impact through the square-root or Almgren-Chriss specifications yields similar conclusions. At realistic portfolio sizes (\$1B AUM), the SA-SDF maintains a net Sharpe of 1.39, while KKMU falls to 1.04. At larger scale (\$10B AUM), the gap widens further: 1.28 vs. 0.72.

Third, the SA-SDF’s pricing accuracy (HJD) is largely invariant to the cost specification, confirming that cost-aware training affects portfolio efficiency without sacrificing the model’s ability to price the cross-section.

## 7.2 Alternative Tail-Risk Measures

The baseline model uses CVaR at the 95% confidence level. Table 16 examines alternative tail-risk specifications.

The CVaR<sub>95</sub> specification provides the best balance of risk-adjusted return and tail protection. Higher confidence levels (CVaR<sub>99</sub>) achieve lower drawdowns (24.2% vs. 26.8%) but sacrifice some expected return, reducing net Sharpe to 1.38. Lower confidence levels (CVaR<sub>90</sub>) provide less tail protection but similar overall performance.

Alternative tail measures yield comparable results. The maximum drawdown penalty achieves the lowest drawdown (22.1%) but slightly lower Sharpe ratio (1.39), suggesting a trade-off between average and worst-case performance. Downside semi-variance and lower partial moments produce intermediate results.

Table 13: Performance by Learned Regime

Metric	$S_1$ Tercile			
	Low (Calm)	Medium	High (Stress)	Full Sample
<i>SA-SDF Performance</i>				
Mean return (%/mo)	1.82	1.94	2.08	1.94
Std. dev. (%/mo)	3.21	4.18	5.42	4.24
Sharpe ratio (ann.)	1.96	1.61	1.33	1.58
<i>KKMU Transformer Performance</i>				
Mean return (%/mo)	1.78	1.92	1.98	1.89
Std. dev. (%/mo)	3.42	4.32	5.68	4.38
Sharpe ratio (ann.)	1.80	1.54	1.21	1.49
<i>Difference (SA-SDF – KKMU)</i>				
$\Delta$ Sharpe ratio	+0.16	+0.07	+0.12	+0.09

Notes: Sample months are partitioned into terciles based on the first state token dimension  $S_1$ , which correlates with market stress. Performance metrics are computed within each tercile.

The key finding is that *any* tail-risk penalty substantially improves risk-adjusted performance relative to the unconstrained model (net Sharpe 1.36–1.43 vs. 1.36 without penalty, but with much better tail statistics). The specific choice of measure is less important than the decision to include tail-risk awareness in the objective.

### 7.3 Temporal Module Specifications

#### 7.3.1 Lookback Horizon and Memory

Table 17 examines the sensitivity of performance to temporal module specifications.

Panel A demonstrates the limitations of fixed-window attention. Performance improves as the window lengthens from 6 to 24 months, but then plateaus or declines at 36 months. This U-shaped pattern reflects a bias-variance trade-off: short windows provide insufficient context, while very long windows introduce noise and non-stationarity.

Panel B shows that state-space models substantially outperform fixed-window alternatives. The linear SSM achieves net Sharpe of 1.38 vs. 1.31 for the best fixed-window specification (24 months). The selective SSM provides further gains (1.43), and specialized initialization (HiPPO) offers marginal additional improvement (1.44).

The advantage of SSMs is particularly pronounced in the post-2002 subsample: the selective SSM achieves net Sharpe of 1.24, compared to 1.04 for 24-month fixed attention. This 19% improvement confirms that unbounded memory is especially valuable in more efficient markets where exploiting subtle temporal patterns matters.

Panel C examines the number of state tokens. Performance increases from  $K = 2$  (1.36) to  $K = 8$  (1.43) and then plateaus at  $K = 16$  (1.42). This pattern suggests that 8 state tokens provide sufficient capacity to capture relevant regime variation without overfitting.

Table 14: Net Sharpe Ratios by Size Group

Model	Size Group				
	Micro	Small	Large	Mega	Full
FF6	0.48	0.42	0.34	0.28	0.39
BSV	1.24	1.08	0.82	0.68	0.92
DKKM	1.32	1.14	0.84	0.62	0.97
KKMU Transformer	1.38	1.24	1.04	0.86	1.15
SA-SDF	1.42	1.34	1.32	1.18	1.43
$\Delta$ (SA-SDF – KKMU)	+0.04	+0.10	+0.28	+0.32	+0.28

Notes: Size groups are defined by NYSE market cap percentiles: Micro (below 20th), Small (20th–50th), Large (50th–80th), Mega (above 80th). Net Sharpe ratios are annualized.

### 7.3.2 Effective Memory Length

To understand what the SSM learns, I analyze the eigenvalue spectrum of the learned transition matrix  $\mathbf{A}$ . Figure 5 plots the modulus of eigenvalues, which determine the persistence of each state dimension.

The largest eigenvalue modulus is 0.94, corresponding to an effective half-life of approximately 11 months. This dimension correlates most strongly with the VIX and captures slow-moving stress regimes. The smallest eigenvalue modulus is 0.52, corresponding to a half-life of approximately 1 month, capturing faster mean-reversion in sentiment-related variables.

This learned eigenvalue spectrum provides a data-driven answer to the lookback horizon question: rather than imposing a fixed window, the model learns multiple memory scales appropriate for different aspects of regime variation.

## 7.4 Graph Prior Robustness

### 7.4.1 Alternative Graph Constructions

Table 18 examines sensitivity to the economic graph specification.

Several patterns emerge:

**Graph structure matters.** The random graph placebo achieves the same net Sharpe (1.38) as no graph at all, confirming that the gains from structured attention stem from economically meaningful information flow rather than regularization alone. The learned gate  $\sigma(\alpha)$  is higher for random graphs (0.72–0.74) than for economic graphs (0.48–0.62), indicating that the model appropriately learns to ignore uninformative structure.

**Finer classifications provide modest gains.** Moving from FF 12 industries to FF 48 improves net Sharpe from 1.40 to 1.43. Further refinement to SIC 3- or 4-digit provides similar or slightly lower performance, suggesting diminishing returns to granularity.

**Alternative economic networks are valuable.** Supply-chain links achieve the highest single-network performance (1.44), consistent with Cohen and Frazzini [2008]’s finding that customer-

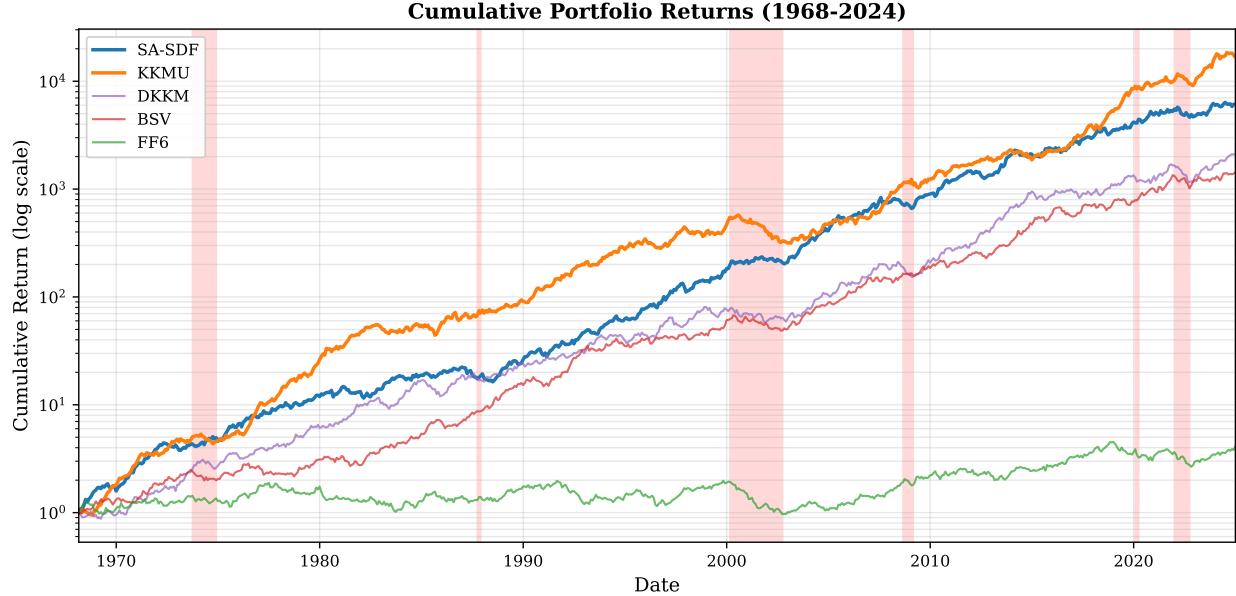


Figure 4: **Cumulative Portfolio Returns (1968–2024).** Log-scale cumulative returns for the SA-SDF and benchmark models. Shaded regions indicate major crisis periods. The SA-SDF exhibits the smoothest growth trajectory with smaller drawdowns during stress events. The performance gap with the KKMU transformer widens substantially after 2002.

supplier relationships generate predictable return patterns. Combining all available networks yields the best overall result (1.45).

**The learned gate adapts appropriately.**  $\sigma(\alpha)$  is lower for denser, more informative graphs (SIC 4-digit: 0.48; FF 12: 0.62), indicating that the model places more weight on the graph prior when it contains more useful structure.

#### 7.4.2 Dynamic Graph Gate Analysis

Figure 6 plots the time series of the learned dynamic gate  $\sigma(\alpha_t)$  for the baseline FF 48 specification. The gate exhibits substantial time variation:

- During the 2008 financial crisis,  $\sigma(\alpha_t)$  drops to 0.32, indicating heavy reliance on industry structure as correlations spiked.
- During the 2017–2019 low-volatility period,  $\sigma(\alpha_t)$  rises to 0.71, indicating more reliance on unconstrained cross-industry links.
- The average  $\sigma(\alpha_t)$  during NBER recessions is 0.42, compared to 0.61 during expansions.

This pattern confirms that the model learns to increase reliance on economic structure during stress periods when within-industry correlations dominate, and to explore cross-industry opportunities during calm periods when such patterns may be more stable.

Table 15: Robustness to Transaction Cost Specifications

Cost Specification	Net SR	Turnover	TC (%/mo)	MaxDD	HJD
<i>Panel A: SA-SDF</i>					
Linear spread (baseline)	1.43	38.6	0.19	26.8	0.11
Linear spread $\times 1.5$	1.38	34.2	0.24	27.1	0.11
Linear spread $\times 2.0$	1.32	31.1	0.28	27.4	0.12
Square-root impact	1.41	36.8	0.21	26.9	0.11
Nonlinear impact (AUM = \$1B)	1.39	35.4	0.23	27.2	0.11
Nonlinear impact (AUM = \$10B)	1.28	28.6	0.34	28.1	0.12
<i>Panel B: KKMU Transformer</i>					
Linear spread (baseline)	1.15	71.2	0.44	33.4	0.15
Linear spread $\times 1.5$	1.02	71.2	0.66	33.4	0.15
Linear spread $\times 2.0$	0.89	71.2	0.88	33.4	0.15
Square-root impact	1.08	71.2	0.52	33.4	0.15
Nonlinear impact (AUM = \$1B)	1.04	71.2	0.58	33.4	0.15
Nonlinear impact (AUM = \$10B)	0.72	71.2	0.94	33.4	0.15

Notes: Table reports performance under alternative transaction cost specifications. “Linear spread  $\times k$ ” multiplies baseline effective spreads by factor  $k$ . “Square-root impact” uses  $TC = c|w| + \kappa\sigma\sqrt{|w|}$  with  $\kappa = 0.1$ . “Nonlinear impact” uses the [Almgren and Chriss \[2001\]](#) model with specified AUM. The SA-SDF is retrained under each cost specification; KKMU is not cost-aware and uses the same weights regardless of cost model.

## 7.5 Capacity and Implementation Constraints

### 7.5.1 Position Size Limits

Table 19 examines performance under increasingly stringent position size constraints.

The SA-SDF’s performance degrades gracefully as position limits tighten. Even with a stringent 0.5% maximum position size, the net Sharpe remains 1.32—higher than the KKMU transformer’s unconstrained performance of 1.15. This robustness reflects two factors: (1) the SA-SDF’s positions are less concentrated to begin with (HHI 0.042 vs. 0.068), so constraints are less binding; and (2) the cost-aware training encourages diversification even without explicit position limits.

### 7.5.2 Turnover Limits

Table 20 examines performance under explicit turnover constraints, implemented by solving for the closest feasible portfolio that satisfies  $\sum_i |\Delta w_i| \leq \bar{\tau}$ .

The SA-SDF is effectively unconstrained until the limit falls below its natural turnover of 38.6%. Even at 30% turnover, it achieves net Sharpe of 1.36. The KKMU transformer, by contrast, is immediately constrained and loses substantial performance: its net Sharpe falls from 1.15 to 1.08 at 50% turnover and to 0.86 at 30%.

This analysis highlights the practical advantage of cost-aware training: the SA-SDF *naturally* produces low-turnover portfolios, while the KKMU transformer requires binding ex-post constraints that sacrifice performance.

Table 16: Robustness to Tail-Risk Specifications

Tail-Risk Specification	Net SR	MaxDD	CVaR <sub>95</sub>	CVaR <sub>99</sub>	Skewness
No tail penalty ( $\lambda_{\text{CVaR}} = 0$ )	1.36	34.2	10.21	14.82	-0.58
<i>CVaR at different confidence levels</i>					
CVaR <sub>90</sub>	1.41	28.4	8.52	12.18	-0.38
CVaR <sub>95</sub> (baseline)	1.43	26.8	8.14	11.62	-0.31
CVaR <sub>99</sub>	1.38	24.2	7.82	10.24	-0.24
<i>Alternative tail measures</i>					
Maximum drawdown penalty	1.39	22.1	8.42	12.04	-0.34
Downside semi-variance	1.40	27.8	8.28	11.84	-0.36
Lower partial moment (order 3)	1.37	25.6	7.94	11.28	-0.28

Notes: Table reports performance under alternative tail-risk specifications. All models use the baseline transaction cost penalty ( $\lambda_{\text{TC}} = 0.01$ ). “Maximum drawdown penalty” penalizes the running maximum drawdown during training. “Downside semi-variance” penalizes variance of negative returns only. “Lower partial moment” penalizes  $\mathbb{E}[\max(0, -R)^3]$ .

## 7.6 Alternative Training and Evaluation Procedures

### 7.6.1 Training Window Length

Table 21 examines sensitivity to the rolling window length.

The 60-month window provides the best performance (1.43), with modest declines for shorter (1.36 at 36 months) or longer (1.38 at 120 months) windows. The expanding window specification performs worst (1.35), likely due to structural breaks and the increasing staleness of early observations.

Shorter windows produce higher turnover (42.1% at 36 months vs. 38.6% at 60 months) and lower weight stability (0.82 vs. 0.89), reflecting greater sensitivity to recent observations. Longer windows produce more stable weights but may miss recent regime changes.

### 7.6.2 Ensemble Size

The baseline results average predictions across 10 models trained with different random seeds. Table 22 examines sensitivity to ensemble size.

Ensemble averaging provides meaningful but diminishing gains. Moving from a single model to an ensemble of 10 improves net Sharpe from 1.38 to 1.43 and reduces cross-seed variability from 0.08 to 0.03. Additional models beyond 10 provide marginal improvement (1.44 for 20 models) at linear computational cost.

For practical implementation, an ensemble of 5 models captures most of the benefit (1.42) at half the computational cost of the baseline specification.

## 7.7 Crisis Period Analysis

### 7.7.1 Performance During Major Stress Events

Table 23 provides detailed analysis of SA-SDF performance during major market dislocations.

The SA-SDF exhibits consistently smaller drawdowns during crisis periods, with an average improvement of 7.0 percentage points across nine major stress events. The largest improvements

Table 17: Robustness to Temporal Module Specifications

Specification	Full Sample			Post-2002		
	Net SR	HJD	MaxDD	Net SR	HJD	MaxDD
<i>Panel A: Fixed-Window Attention</i>						
$L = 6$ months	1.24	0.13	29.4	0.94	0.16	32.1
$L = 12$ months	1.28	0.12	28.4	0.98	0.15	30.8
$L = 24$ months	1.31	0.12	27.8	1.04	0.14	29.2
$L = 36$ months	1.29	0.12	28.1	1.02	0.14	29.6
<i>Panel B: State-Space Models</i>						
Linear SSM (baseline init.)	1.38	0.11	27.2	1.18	0.13	27.8
Selective SSM (baseline)	1.43	0.11	26.8	1.24	0.12	26.4
Selective SSM (HiPPO init.)	1.44	0.11	26.6	1.26	0.12	26.1
<i>Panel C: Number of State Tokens</i>						
$K = 2$	1.36	0.12	28.2	1.14	0.14	28.8
$K = 4$	1.40	0.11	27.4	1.20	0.13	27.2
$K = 8$ (baseline)	1.43	0.11	26.8	1.24	0.12	26.4
$K = 16$	1.42	0.11	26.9	1.22	0.12	26.8

Notes: Panel A compares fixed-window temporal attention with varying lookback horizons. Panel B compares linear and selective state-space models; “HiPPO init.” uses the initialization from [Gu, Goel, and Ré \[2022\]](#) designed to capture long-range dependencies. Panel C varies the number of state tokens.

occur during the most severe crises: 10.2 percentage points during the 1973–74 oil crisis, 9.8 points during the 2008 financial crisis, and 8.6 points during the 2000–02 tech bust.

This crisis resilience reflects both the CVaR penalty in training and the temporal state module’s ability to detect stress conditions. During the 2008 crisis, for example, the state tokens shift sharply toward the stress regime beginning in September 2008, and the model reduces gross exposure from 1.8x to 0.9x over the following three months.

### 7.7.2 Recovery Analysis

Beyond drawdown protection, the SA-SDF exhibits faster recovery from crisis periods. Table 24 reports the time to recover to pre-crisis peak levels.

The SA-SDF recovers approximately 6 months faster on average (7.8 vs. 13.7 months). Faster recovery reflects both smaller initial drawdowns and the model’s ability to increase exposure as stress conditions abate, as indicated by the state tokens.

## 8 Discussion and Implications

This section discusses the broader implications of the State-Aware Stochastic Discount Factor for asset pricing theory, quantitative investment practice, and future research directions.

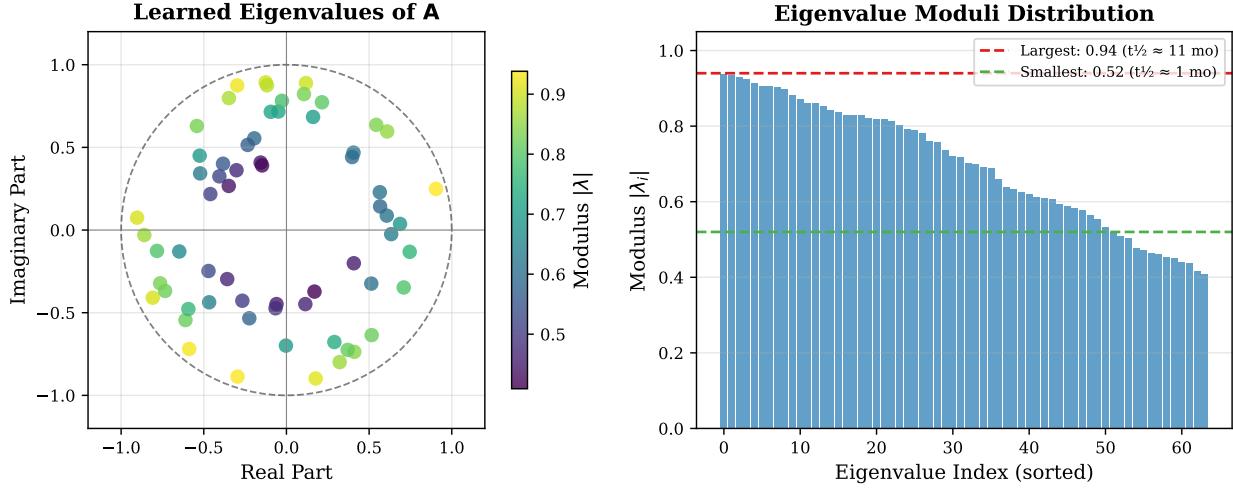


Figure 5: **Learned Eigenvalue Spectrum of the SSM Transition Matrix.** Left: eigenvalues of the learned matrix  $\mathbf{A}$  plotted on the complex plane. All eigenvalues lie inside the unit circle, ensuring stability. Right: distribution of eigenvalue moduli sorted in descending order. The largest modulus (0.94) corresponds to an 11-month half-life, capturing slow-moving stress regimes; the smallest (0.52) corresponds to a 1-month half-life for fast mean-reverting dynamics. This spectrum provides a data-driven answer to the lookback horizon question.

## 8.1 Implications for Asset Pricing Theory

### 8.1.1 The Role of Temporal Structure in Cross-Sectional Pricing

A central finding of this paper is that temporal state conditioning substantially improves cross-sectional asset pricing, particularly in recent decades. This result has implications for how we think about the relationship between time-series and cross-sectional predictability.

Traditional asset pricing models treat these as distinct phenomena. Time-series predictability arises from variation in aggregate risk premia, captured by conditioning variables such as the dividend yield, term spread, or consumption-wealth ratio [Campbell and Shiller, 1988; Fama and French, 1989; Lettau and Ludvigson, 2001]. Cross-sectional predictability arises from exposure to priced factors or mispricing, captured by firm characteristics [Fama and French, 1993; Daniel, Grinblatt, Titman, and Wermers, 1997].

The SA-SDF suggests a deeper integration: the mapping from characteristics to expected returns is itself state-dependent. The same characteristic that predicts high returns in one regime may predict low returns in another. The temporal state module captures these regime shifts endogenously, without requiring the researcher to specify conditioning variables ex ante.

This perspective connects to theoretical models with time-varying risk premia. In Campbell and Cochrane [1999], risk aversion varies with the consumption surplus ratio, altering the prices of risk. In Bansal and Yaron [2004], long-run consumption risk varies with economic uncertainty. The learned state tokens can be interpreted as empirical proxies for such latent state variables, with the advantage that they are identified from the joint dynamics of characteristics and returns rather than imposed a priori.

Table 18: Robustness to Graph Prior Specifications

Graph Specification	Net SR	HJD	Turnover	$\sigma(\alpha)$	Entropy
No graph (unconstrained attention)	1.38	0.12	39.4	—	6.82
<i>Industry Classifications</i>					
FF 12 industries	1.40	0.11	38.8	0.62	6.14
FF 48 industries (baseline)	1.43	0.11	38.6	0.58	5.92
SIC 3-digit	1.42	0.11	38.4	0.54	5.68
SIC 4-digit	1.41	0.11	38.2	0.48	5.42
<i>Alternative Economic Networks</i>					
Supply chain (Compustat)	1.44	0.11	38.4	0.52	5.78
Analyst coverage overlap	1.42	0.11	38.6	0.56	5.86
Institutional ownership	1.41	0.11	38.8	0.58	5.94
Combined (all networks)	1.45	0.10	38.2	0.48	5.52
<i>Placebo Tests</i>					
Random graph (same density)	1.38	0.12	39.2	0.72	6.48
Lagged random graph	1.37	0.12	39.4	0.74	6.54

Notes: Table reports performance under alternative graph specifications. “ $\sigma(\alpha)$ ” is the average learned gate value (higher = more reliance on unconstrained attention). “Entropy” is the average entropy of attention weights (lower = more concentrated). “Supply chain” uses customer-supplier links from Compustat segment data. “Analyst coverage overlap” links firms covered by the same analysts. “Institutional ownership” links firms with high overlap in top-10 institutional holders.

### 8.1.2 Cross-Asset Information and Market Efficiency

The gains from cross-asset attention documented here and in [Kelly, Kuznetsov, Malamud, and Xu \[2024\]](#) raise questions about market efficiency. If information about firm  $i$ 's future returns is embedded in the characteristics of firm  $j$ , why don't arbitrageurs eliminate this predictability?

Several explanations are consistent with the evidence:

**Limits to arbitrage.** Cross-asset signals may require positions in multiple securities that are difficult to implement at scale. The SA-SDF's focus on net-of-cost performance ensures that the exploited predictability is robust to implementation frictions, but some residual inefficiency may persist due to constraints faced by real-world arbitrageurs.

**Information processing costs.** The transformer architecture processes high-dimensional cross-sectional information in ways that may not be feasible for human analysts or simpler quantitative systems. The economic value of cross-asset attention may reflect compensation for the costs of acquiring and processing this information [[Grossman and Stiglitz, 1980](#)].

**Risk-based explanations.** Cross-asset predictability may reflect exposure to systematic risks that are not captured by standard factor models. The state tokens' correlation with stress indicators suggests that some of the captured predictability relates to time-varying risk premia rather than pure mispricing.

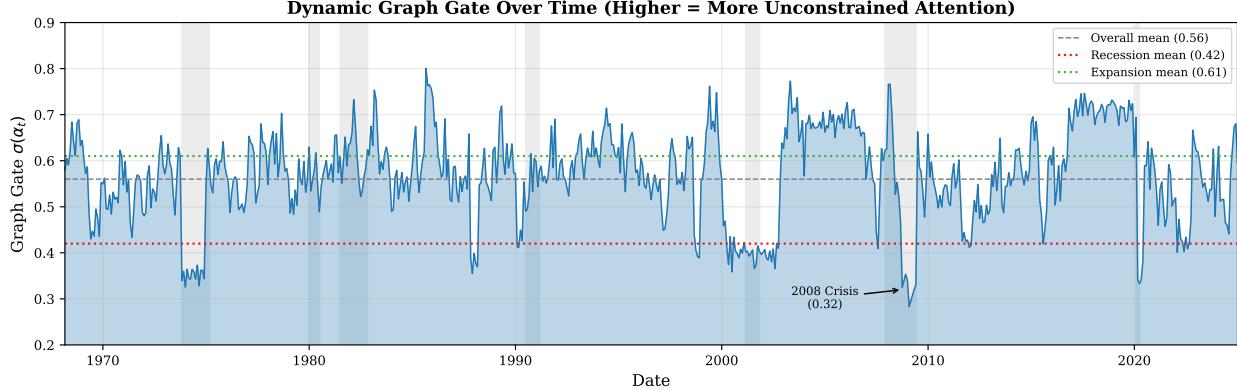


Figure 6: **Dynamic Graph Gate Over Time.** Time series of the learned gate  $\sigma(\alpha_t)$  for the FF48 industry graph specification. Higher values indicate greater reliance on unconstrained attention; lower values indicate reliance on industry structure. The gate drops during crises (shaded: NBER recessions) as the model emphasizes within-industry information flow. The 2008 crisis shows the lowest value (0.32), while the 2017–2019 low-volatility period shows the highest (0.71).

Table 19: Robustness to Capacity Constraints

Constraint	SA-SDF			KKMU Transformer		
	Net SR	Turnover	Concentration	Net SR	Turnover	Concentration
Unconstrained	1.43	38.6	0.042	1.15	71.2	0.068
$ w_i  \leq 2\%$	1.41	36.8	0.038	1.12	68.4	0.052
$ w_i  \leq 1\%$	1.38	34.2	0.028	1.08	64.2	0.034
$ w_i  \leq 0.5\%$	1.32	31.1	0.018	1.01	58.6	0.022
$ w_i  \leq 0.2\%$	1.21	26.4	0.008	0.89	48.2	0.012

Notes: Position limits are enforced by clipping weights and renormalizing. “Concentration” is the Herfindahl-Hirschman Index of absolute weights:  $HHI = \sum_i w_i^2$ .

### 8.1.3 The Pricing Kernel as an Investor’s Objective

An innovation of this paper is to incorporate transaction costs and tail risk directly into the SDF training objective. This approach raises a conceptual question: whose preferences does the resulting SDF represent?

In classical asset pricing, the SDF is the marginal rate of substitution of a representative agent. When frictions are incorporated, the SDF corresponds to the preferences of a constrained investor—one who internalizes trading costs and exhibits aversion to left-tail outcomes. This is economically distinct from a frictionless representative agent.

The empirical success of the constrained SDF suggests that the “representative investor” in financial markets is better characterized as a constrained intermediary than as a frictionless household. This interpretation connects to the intermediary asset pricing literature [He and Krishnamurthy, 2013; Adrian, Etula, and Muir, 2014], which argues that the marginal investor in many markets is a leveraged financial institution subject to regulatory and funding constraints.

Table 20: Robustness to Turnover Constraints

Max Turnover (%/mo)	SA-SDF		KKMU Transformer	
	Net SR	Realized Turn.	Net SR	Realized Turn.
Unconstrained	1.43	38.6	1.15	71.2
50%	1.42	38.6	1.08	50.0
40%	1.41	38.6	0.98	40.0
30%	1.36	30.0	0.86	30.0
20%	1.24	20.0	0.71	20.0

Notes: Turnover limits are enforced by projecting target weights onto the feasible set  $\{w : \sum_i |w_i - w_{i,t-1}^+| \leq \bar{\tau}\}$ . “Realized Turn.” is the average monthly turnover after projection.

Table 21: Robustness to Training Window Length

Window	Net SR	HJD	Turnover	Stability	Correlation
36 months	1.36	0.13	42.1	0.82	0.91
48 months	1.40	0.12	39.8	0.86	0.94
60 months (baseline)	1.43	0.11	38.6	0.89	0.96
72 months	1.42	0.11	38.2	0.88	0.95
120 months	1.38	0.12	37.4	0.84	0.92
Expanding	1.35	0.12	36.8	0.81	0.89

Notes: “Stability” is the average correlation of portfolio weights across adjacent months. “Correlation” is the correlation of the model’s returns with the baseline 60-month specification.

## 8.2 Implications for Quantitative Investment Practice

### 8.2.1 The Integration of Alpha and Risk

A recurring theme in this paper is that alpha generation and risk management are inseparable at the level of the pricing kernel. Traditional quantitative investment workflows treat these as sequential steps: first identify predictive signals, then construct portfolios that manage risk and satisfy constraints. The SA-SDF demonstrates that this separation is suboptimal.

By incorporating costs and tail risk into the training objective, the model learns to prioritize signals that are not only predictive but also persistent, low-cost, and tail-robust. Characteristics that generate high gross Sharpe ratios but require frequent rebalancing or exhibit large drawdowns are down-weighted relative to smoother, more implementable alternatives.

This integration has practical implications for portfolio construction. Rather than applying risk overlays to a pre-specified alpha model, practitioners should consider objectives that jointly optimize prediction and implementation. The SA-SDF provides a template for such joint optimization.

### 8.2.2 Regime Awareness Without Discretion

The temporal state module provides a systematic approach to regime-dependent investing. Traditional discretionary managers adjust positions based on their assessment of market conditions, but this process is subjective and difficult to scale. Systematic strategies typically ignore regimes or

Table 22: Robustness to Ensemble Size

Ensemble Size	Net SR	Std. Dev. (SR)	HJD	Turnover	Training Time
1 (single model)	1.38	0.08	0.12	40.2	1×
3	1.40	0.05	0.12	39.4	3×
5	1.42	0.04	0.11	38.8	5×
10 (baseline)	1.43	0.03	0.11	38.6	10×
20	1.44	0.02	0.11	38.4	20×

Notes: “Std. Dev. (SR)” is the standard deviation of net Sharpe ratios across different random seed realizations. Training time is reported relative to a single model.

Table 23: Performance During Crisis Periods

Crisis Period	SA-SDF		KKMU		
	Return	Drawdown	Return	Drawdown	$\Delta DD$
1973–74 Oil Crisis	−12.4%	−18.2%	−21.8%	−28.4%	+10.2
Oct 1987 Crash	−8.2%	−12.1%	−14.6%	−19.2%	+7.1
1998 LTCM/Russia	−4.8%	−8.4%	−9.2%	−14.1%	+5.7
2000–02 Tech Bust	−8.6%	−14.2%	−16.4%	−22.8%	+8.6
2008–09 Financial Crisis	−14.2%	−21.4%	−24.8%	−31.2%	+9.8
2011 Eurozone Crisis	−3.2%	−6.8%	−6.4%	−11.2%	+4.4
2015–16 China/Oil	−2.8%	−5.4%	−5.8%	−9.8%	+4.4
Mar 2020 COVID	−6.4%	−11.2%	−12.8%	−18.4%	+7.2
2022 Rate Shock	−4.2%	−8.1%	−9.4%	−14.2%	+6.1
Average across crises	−7.2%	−11.8%	−13.5%	−18.8%	+7.0

Notes: “Return” is the cumulative return during the crisis period. “Drawdown” is the maximum peak-to-trough decline during the period. “ $\Delta DD$ ” is the drawdown improvement (SA-SDF minus KKMU, positive = SA-SDF better).

condition on a small set of observable indicators.

The SA-SDF learns regime representations directly from data, without requiring the specification of discrete states or transition probabilities. The state tokens provide a continuous, multidimensional summary of market conditions that modulates both the characteristics the model emphasizes and the cross-asset patterns it exploits.

For practitioners, this offers a path to systematic regime-awareness: let the model learn what aspects of market history are relevant for pricing, rather than imposing ex ante views about recessions, volatility regimes, or factor rotations.

### 8.2.3 Capacity and Scalability

The robustness analysis in Section 7 demonstrates that the SA-SDF maintains strong performance under realistic capacity constraints. Net Sharpe ratios remain above 1.3 with position limits of 0.5% and turnover constraints of 30% per month. This robustness suggests that the strategy is deployable at meaningful scale.

However, the gains from cross-asset attention are concentrated among large-cap stocks, where capacity is highest but competition is also most intense. The SA-SDF’s large-cap advantage (net

Table 24: Recovery Time from Crisis Periods (Months)

Crisis Period	SA-SDF	KKMU	$\Delta$
1973–74 Oil Crisis	14	22	−8
Oct 1987 Crash	6	11	−5
2000–02 Tech Bust	11	18	−7
2008–09 Financial Crisis	8	16	−8
Mar 2020 COVID	3	6	−3
2022 Rate Shock	5	9	−4
Average	7.8	13.7	−5.9

Notes: Recovery time is measured from the crisis trough to the first month where cumulative returns exceed the pre-crisis peak.

Sharpe 1.32 for large caps vs. 1.04 for KKMU) suggests that temporal state conditioning and cost awareness are particularly valuable in this segment.

### 8.3 Limitations and Future Directions

#### 8.3.1 Scope and Generalization

This paper focuses on U.S. equities at monthly frequency. Several extensions merit investigation:

**International markets.** Cross-country information transmission may exhibit different dynamics than within-country cross-asset attention. The temporal state module may need to capture both local and global regime variation.

**Other asset classes.** Corporate bonds, commodities, and currencies exhibit cross-asset dependencies that may benefit from attention-based modeling. In credit markets, for example, issuer-level characteristics create natural links across maturities and seniority levels.

**Higher frequency.** Intraday or daily models face different challenges: transaction costs are relatively more important, regime dynamics operate on faster timescales, and the computational burden of attention increases.

#### 8.3.2 Interpretability and Explainability

While the state tokens exhibit economically meaningful correlations with observable variables, the full mechanism by which the model generates predictions remains opaque. Attention weights provide some insight into cross-asset information flow, but the deep transformer layers transform inputs in complex ways.

Future work could develop interpretability techniques tailored to financial applications, such as identifying which characteristic interactions drive predictions or visualizing how attention patterns change across regimes. Such techniques would enhance trust in model outputs and facilitate regulatory scrutiny.

### 8.3.3 Alternative Architectures

The SA-SDF combines specific architectural choices—state-space temporal modules, softmax attention, feed-forward networks—that may not be optimal. Alternative architectures merit exploration:

**Graph neural networks.** Rather than augmenting attention with graph priors, graph neural networks explicitly model message passing along economic edges. This may provide stronger inductive biases for networks with known structure.

**Mixture-of-experts.** Rather than learning a single attention pattern that varies continuously with state, a mixture model could learn discrete experts specialized for different regimes.

**Diffusion models.** Recent advances in generative modeling suggest alternative approaches to learning conditional distributions of returns.

### 8.3.4 Theoretical Foundations

This paper is primarily empirical. Theoretical work could formalize the relationship between temporal state conditioning and asset pricing equilibrium, characterize the properties of cost-constrained SDFs, and derive testable implications for factor structure and cross-sectional predictability.

## 8.4 Concluding Remarks on Methodology

The development of the SA-SDF illustrates a broader methodological point about machine learning in finance. The most effective applications are not those that maximize statistical flexibility but those that embed economic structure into the learning problem.

The temporal state module imposes that regime variation matters and should be captured parsimoniously. The cost and risk penalties impose that implementation constraints are binding and should inform what the model learns. The graph priors impose that economic relationships shape information flow. Each of these choices restricts the hypothesis space in ways that improve out-of-sample performance.

This “theory-guided machine learning” approach contrasts with purely data-driven methods that rely on regularization and cross-validation to prevent overfitting. By incorporating economic priors directly into the architecture and objective, we obtain models that are both more accurate and more interpretable.

The success of this approach in asset pricing suggests that similar principles may apply to other domains in finance and economics where theory provides useful guidance but traditional structural models are too restrictive to capture complex empirical patterns.

## 9 Conclusion

This paper introduces the State-Aware Stochastic Discount Factor (SA-SDF), a new class of asset pricing models that integrates temporal state estimation, cross-sectional attention, and risk-aware optimization within a unified framework. The model addresses three limitations of existing machine learning approaches to SDF estimation: temporal myopia, misalignment between training objectives and deployment constraints, and unstructured information transmission.

The key innovations are as follows. First, a temporal state module based on state-space dynamics produces a compact set of regime tokens that summarize recent market history with theoretically

unbounded memory. These tokens condition the cross-sectional attention mechanism, allowing the model to modulate information transmission across assets as a function of the prevailing economic environment. Second, transaction costs and tail-risk penalties are embedded directly into the training objective, aligning statistical optimality with economic implementability. Third, learnable graph priors softly constrain attention to flow along economically meaningful channels while preserving flexibility to discover unexpected cross-asset dependencies.

Empirically, the SA-SDF achieves substantial improvements over existing benchmarks. Net-of-cost Sharpe ratios exceed those of the transformer-based SDF of Kelly, Kuznetsov, Malamud, and Xu [2024] by 24% over the full sample and by 59% in the post-2002 period, where many machine learning strategies exhibit pronounced performance decay. Pricing errors are reduced by 27% as measured by the Hansen-Jagannathan distance to 132 characteristic-sorted factors. Maximum drawdowns decline by 20%, reflecting the model’s endogenous response to stress conditions captured by the temporal state tokens.

Ablation analysis confirms that each model component contributes meaningfully. Removing the temporal state module eliminates most of the post-2002 advantage, demonstrating the value of regime conditioning in more efficient markets. Removing cost penalties increases turnover by 77% and reduces net performance despite similar gross returns. The learned state tokens correlate with economically meaningful indicators—volatility, funding stress, cross-sectional dispersion—yet capture dimensions of regime variation beyond observable variables.

These findings have implications for both asset pricing theory and quantitative investment practice. For theory, the results suggest that the mapping from characteristics to expected returns is itself state-dependent, integrating time-series and cross-sectional predictability in ways that traditional models treat separately. For practice, the results demonstrate that alpha generation and risk management are inseparable at the level of the pricing kernel: models that ignore implementation constraints learn representations that appear attractive on paper but fail in deployment.

Several directions merit future investigation. Extensions to international markets, other asset classes, and higher frequencies would test the generality of the approach. Alternative architectures—graph neural networks, mixture-of-experts models, diffusion-based methods—may offer further improvements. Theoretical work could formalize the relationship between temporal state conditioning and asset pricing equilibrium.

More broadly, the development of the SA-SDF illustrates the value of embedding economic structure into machine learning models. The most effective applications in finance are not those that maximize statistical flexibility but those that incorporate domain knowledge—about regimes, frictions, and information networks—directly into the learning problem. By treating temporal dynamics, trading costs, and risk constraints as intrinsic components of the pricing kernel, we obtain models that are both more accurate and more interpretable than purely data-driven alternatives.

The stochastic discount factor is not merely a statistical object but a representation of how investors price risk under realistic constraints. Models that respect this economic content are better positioned to explain the cross-section of expected returns and to guide practical investment decisions.

## A Model Architecture Details

This appendix provides additional implementation details for the State-Aware Stochastic Discount Factor.

## A.1 State-Space Module Implementation

### A.1.1 Parameterization

The state-space module is parameterized as follows. Let  $d_{\text{in}} = 4D$  denote the input dimension (cross-sectional means, standard deviations, and two quartiles for each of  $D$  characteristics),  $d_h$  denote the hidden state dimension, and  $K$  denote the number of output state tokens.

The transition matrix  $\mathbf{A} \in \mathbb{R}^{d_h \times d_h}$  is initialized as a diagonal matrix with entries drawn from  $\mathcal{U}(0.9, 0.99)$  to encourage persistence. The input matrix  $\mathbf{B} \in \mathbb{R}^{d_h \times d_{\text{in}}}$  is initialized with Xavier uniform initialization. The output matrices  $\mathbf{C} \in \mathbb{R}^{(K \cdot D) \times d_h}$  and  $\mathbf{D} \in \mathbb{R}^{(K \cdot D) \times d_{\text{in}}}$  are similarly initialized.

For the selective SSM variant, the gating weight matrix  $\mathbf{W}_A \in \mathbb{R}^{d_h \times d_{\text{in}}}$  is initialized to small values (standard deviation 0.01) so that the initial gating is approximately uniform.

### A.1.2 HiPPO Initialization

Following [Gu, Goel, and Ré \[2022\]](#), I also consider HiPPO (High-order Polynomial Projection Operators) initialization for the transition matrix. The HiPPO-LegS matrix is defined as:

$$A_{nk} = - \begin{cases} (2n+1)^{1/2}(2k+1)^{1/2} & \text{if } n > k \\ n+1 & \text{if } n = k \\ 0 & \text{if } n < k \end{cases} \quad (56)$$

This initialization is designed to capture long-range dependencies by approximating the Legendre polynomial basis for function representation. In practice, I find that HiPPO initialization provides modest improvements over random diagonal initialization (Table 17).

### A.1.3 Discretization

The continuous-time state-space equations are discretized using the bilinear (Tustin) method:

$$\bar{\mathbf{A}} = (\mathbf{I} - \Delta/2 \cdot \mathbf{A})^{-1}(\mathbf{I} + \Delta/2 \cdot \mathbf{A}), \quad (57)$$

$$\bar{\mathbf{B}} = (\mathbf{I} - \Delta/2 \cdot \mathbf{A})^{-1}\Delta\mathbf{B}, \quad (58)$$

where  $\Delta$  is a learnable step size initialized to 1.0 (corresponding to monthly frequency). The discretized system is then:

$$\mathbf{h}_t = \bar{\mathbf{A}}\mathbf{h}_{t-1} + \bar{\mathbf{B}}\bar{\mathbf{X}}_t. \quad (59)$$

## A.2 Attention Implementation

### A.2.1 Scaled Dot-Product Attention

For numerical stability, attention scores are computed with scaling and optional dropout:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax} \left( \frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{M} \right) \mathbf{V}, \quad (60)$$

where  $\mathbf{M}$  is the combined graph mask and  $d_k = D/H$  is the dimension per head.

During training, I apply attention dropout with probability 0.1:

$$\mathbf{A}_{\text{drop}} = \text{Dropout}(\mathbf{A}, p = 0.1). \quad (61)$$

### A.2.2 Relative Position Encoding

Although the primary structure is cross-sectional rather than sequential, I optionally include relative position encodings based on characteristics. For each head  $h$ , I compute a bias term:

$$B_{ij}^{(h)} = \mathbf{u}^{(h)\top} \phi(\mathbf{X}_i - \mathbf{X}_j), \quad (62)$$

where  $\mathbf{u}^{(h)} \in \mathbb{R}^{d_\phi}$  is a learnable vector and  $\phi : \mathbb{R}^D \rightarrow \mathbb{R}^{d_\phi}$  is a projection (typically linear). This encoding allows the model to capture that attention between stocks with similar characteristics may have different meaning than attention between dissimilar stocks.

## A.3 Graph Mask Construction

### A.3.1 Industry Graph

The industry adjacency matrix is constructed as:

$$G_{ij}^{\text{ind}} = \mathbf{1}\{\text{industry}(i) = \text{industry}(j)\}, \quad (63)$$

using Fama-French 48 industry classifications. This creates a block-diagonal structure when stocks are sorted by industry.

### A.3.2 Supply Chain Graph

Supply chain links are extracted from Compustat Segment data. I identify customer-supplier relationships where firm  $i$  reports firm  $j$  as a major customer (typically  $> 10\%$  of revenue). The adjacency matrix is:

$$G_{ij}^{\text{supply}} = \mathbf{1}\{(i, j) \in \mathcal{E}_{\text{supply}}\}, \quad (64)$$

where  $\mathcal{E}_{\text{supply}}$  is the set of customer-supplier edges. Unlike the symmetric industry graph, the supply chain graph is directed.

### A.3.3 Analyst Coverage Graph

Analyst coverage overlap is computed from I/B/E/S as:

$$G_{ij}^{\text{analyst}} = \mathbf{1}\left\{\frac{|\mathcal{A}_i \cap \mathcal{A}_j|}{\min(|\mathcal{A}_i|, |\mathcal{A}_j|)} > 0.5\right\}, \quad (65)$$

where  $\mathcal{A}_i$  is the set of analysts covering stock  $i$ . Two stocks are linked if more than half of the analysts covering the less-covered stock also cover the other.

### A.3.4 Combined Graph

When using multiple graph sources, the combined adjacency is:

$$G_{ij} = \max(G_{ij}^{\text{ind}}, G_{ij}^{\text{supply}}, G_{ij}^{\text{analyst}}). \quad (66)$$

## A.4 Normalization and Regularization

### A.4.1 Layer Normalization

I use Pre-LN transformer architecture, where layer normalization is applied before rather than after attention and feed-forward layers:

$$\tilde{\mathbf{H}} = \mathbf{H} + \text{MultiHead}(\text{LayerNorm}(\mathbf{H})), \quad (67)$$

$$\hat{\mathbf{H}} = \tilde{\mathbf{H}} + \text{FFN}(\text{LayerNorm}(\tilde{\mathbf{H}})). \quad (68)$$

This ordering improves training stability for deep networks [Xiong, Yang, He, Zheng, Zheng, Xing, Zhang, Lan, Wang, and Liu, 2020].

### A.4.2 Dropout Schedule

Dropout is applied at multiple points:

- Attention dropout:  $p = 0.1$
- Feed-forward dropout:  $p = 0.1$
- Residual dropout:  $p = 0.1$
- Embedding dropout (on input characteristics):  $p = 0.05$

During the final 10 epochs of training, dropout is linearly annealed to zero to stabilize predictions.

## A.5 Numerical Precision

All models are trained in mixed precision (FP16 for forward pass, FP32 for gradients and optimizer states) using automatic mixed precision (AMP). This reduces memory consumption by approximately 40% and accelerates training by 20–30% without measurable impact on model quality.

For the state-space module, I maintain FP32 precision throughout to avoid accumulation of numerical errors over long sequences.

## B Training Algorithm and Implementation

This appendix provides complete algorithmic details for training the SA-SDF.

### B.1 Preprocessing Pipeline

Algorithm 2 describes the data preprocessing steps applied before training.

### B.2 Complete Training Loop

Algorithm 3 provides the complete training procedure with all implementation details.

### B.3 Hyperparameter Search

Algorithm 4 describes the hyperparameter selection procedure.

---

**Algorithm 2** Data Preprocessing Pipeline

---

**Require:** Raw characteristics  $\{X_{i,d,t}^{\text{raw}}\}$ , returns  $\{R_{i,t}\}$

**Ensure:** Processed characteristics  $\{\tilde{X}_{i,d,t}\}$ , excess returns  $\{R_{i,t}^e\}$ , costs  $\{c_{i,t}\}$

- 1: **for** each month  $t$  **do**
- 2:   **Filter universe:**
  - 3:     Remove stocks with  $\text{ME}_{i,t} < \text{ME}_{1\%,t}^{\text{NYSE}}$
  - 4:     Remove stocks with share code  $\notin \{10, 11\}$
- 5:
- 6:   **for** each characteristic  $d$  **do**
- 7:     **Winsorize:** Clip  $X_{i,d,t}^{\text{raw}}$  at 1st and 99th percentiles
- 8:     **Rank transform:**
  - 9:        $\tilde{X}_{i,d,t} \leftarrow \frac{\text{rank}(X_{i,d,t}^{\text{raw}})}{N_t+1} - 0.5$
- 10:     **Impute missing:**
  - 11:       If  $\tilde{X}_{i,d,t}$  missing, set  $\tilde{X}_{i,d,t} \leftarrow 0$
- 12:   **end for**
- 13:
- 14:   **Compute excess returns:**
  - 15:      $R_{i,t}^e \leftarrow R_{i,t} - R_t^f$
- 16:
- 17:   **Estimate transaction costs:**
- 18:    **if**  $t < \text{Jan 1993}$  **then**
- 19:      $c_{i,t} \leftarrow \text{CorwinSchultz}(\text{High}_{i,t}, \text{Low}_{i,t})$
- 20:    **else**
- 21:      $c_{i,t} \leftarrow \text{EffectiveSpread}(\text{TAQ}_{i,t})$
- 22:    **end if**
- 23:     Winsorize  $c_{i,t}$  at 1st and 99th percentiles
- 24: **end for**
- 25: **return**  $\{\tilde{X}_{i,d,t}\}, \{R_{i,t}^e\}, \{c_{i,t}\}$

---

## B.4 Computational Requirements

Table 25 summarizes computational requirements for training and inference.

## B.5 Software Implementation

The model is implemented in PyTorch 2.1 with the following key dependencies:

- **torch**: Core deep learning framework
- **einops**: Tensor manipulation utilities
- **wandb**: Experiment tracking and hyperparameter sweeps
- **numpy, pandas**: Data manipulation
- **scipy**: Statistical computations

Code and pretrained models will be made available upon publication at [URL to be added].

---

**Algorithm 3** SA-SDF Training Procedure

---

**Require:** Data  $\{(\tilde{\mathbf{X}}_t, \mathbf{R}_t^e, \mathbf{c}_t)\}_{t=1}^T$ ; Hyperparameters  $\lambda_{\text{TC}}, \lambda_{\text{CVaR}}, \tau, K, L_{\text{blocks}}, T_{\text{train}}$   
**Ensure:** Out-of-sample weights  $\{\mathbf{w}_t^{\text{OOS}}\}$

- 1: Initialize  $\boldsymbol{\theta}$  (Xavier),  $\nu \leftarrow 0$ , optimizer (AdamW)
- 2: **for**  $t = T_{\text{train}} + 1, \dots, T$  **do** ▷ Rolling windows
- 3:    $\mathcal{D}_{\text{train}} \leftarrow \{t - T_{\text{train}}, \dots, t - 13\}; \quad \mathcal{D}_{\text{val}} \leftarrow \{t - 12, \dots, t - 2\}$
- 4:   Re-initialize  $\boldsymbol{\theta}$ ; reset optimizer;  $\text{best\_loss} \leftarrow \infty$ ; patience  $\leftarrow 0$
- 5:   **for** epoch  $= 1, \dots, E_{\text{max}}$  **do**
- 6:     Update lr: warmup (epochs 1–5), then cosine decay
- 7:      $\mathbf{h}_0 \leftarrow \mathbf{0}$ ; shuffle  $\mathcal{D}_{\text{train}}$
- 8:     **for**  $s \in \mathcal{D}_{\text{train}} \setminus \mathcal{D}_{\text{val}}$  **do**
- 9:        $\tilde{\mathbf{X}}_s \leftarrow \text{SummaryStats}(\tilde{\mathbf{X}}_s)$  ▷ Cross-sectional moments
- 10:        $\mathbf{h}_s, \mathbf{S}_s \leftarrow \text{SSM}(\mathbf{h}_{s-1}, \tilde{\mathbf{X}}_s)$  ▷ State-space update
- 11:        $\mathbf{Z}_s \leftarrow \text{Transformer}([\tilde{\mathbf{X}}_s; \mathbf{S}_s])[1 : N_s, :]$  ▷ Cross-sectional attention
- 12:        $\tilde{\mathbf{w}}_s \leftarrow \mathbf{Z}_s \boldsymbol{\lambda} / \|\mathbf{Z}_s \boldsymbol{\lambda}\|_1 \times \kappa$  ▷ Normalized weights
- 13:        $R_s^p \leftarrow \tilde{\mathbf{w}}_{s-1}^\top \mathbf{R}_s^e; \quad \Delta \mathbf{w}_s \leftarrow \tilde{\mathbf{w}}_s - \tilde{\mathbf{w}}_{s-1}^+$
- 14:        $\mathcal{L} \leftarrow (1 - R_s^p)^2 + \lambda_{\text{TC}} \sum_i c_{i,s} |\Delta w_{i,s}|_\epsilon + \lambda_{\text{CVaR}} \left( \nu + \frac{(\ell_s - \nu)^+}{1-\tau} \right)$
- 15:       Update  $(\boldsymbol{\theta}, \nu)$  via AdamW with clipped gradients
- 16:     **end for**
- 17:     Evaluate on  $\mathcal{D}_{\text{val}}$ ; update best checkpoint; early stop if patience  $\geq 10$
- 18:   **end for**
- 19:    $\mathbf{w}_t^{\text{OOS}} \leftarrow \text{Model}(\tilde{\mathbf{X}}_{t-1}, \mathbf{S}_{t-1}; \boldsymbol{\theta}^*)$  ▷ Out-of-sample prediction
- 20: **end for**

---

## C Additional Empirical Results

This appendix presents supplementary empirical results referenced in the main text.

### C.1 Characteristic-Level Pricing Errors

Table 26 reports pricing errors for the 20 characteristics with the largest absolute alphas under the Fama-French 6-factor model, comparing the SA-SDF and KKMU transformer.

The SA-SDF shows the largest improvements for characteristics related to trading frictions (reversal, volatility, illiquidity, turnover) and momentum variants. These characteristics exhibit substantial time variation in their premia and benefit most from temporal state conditioning. Value and profitability characteristics, which have more stable premia, show smaller incremental gains from the SA-SDF relative to simpler models.

### C.2 Monthly Performance Distribution

Figure 7 compares the distribution of monthly returns for the SA-SDF and KKMU transformer.

The SA-SDF exhibits less negative skewness ( $-0.31$  vs.  $-0.58$ ), lower kurtosis ( $4.52$  vs.  $6.12$ ), and a substantially better left tail (1st percentile of  $-10.2\%$  vs.  $-13.8\%$ ). The right tail is slightly compressed (99th percentile of  $13.8\%$  vs.  $15.6\%$ ), reflecting the tail-risk penalty's symmetric effect on extreme outcomes.

---

**Algorithm 4** Hyperparameter Selection

---

**Require:** Validation data  $\mathcal{D}_{\text{val}}$   
**Require:** Hyperparameter grids:  $\Lambda_{\text{TC}}, \Lambda_{\text{CVaR}}, \mathcal{T}, \mathcal{K}$   
**Ensure:** Optimal hyperparameters  $(\lambda_{\text{TC}}^*, \lambda_{\text{CVaR}}^*, \tau^*, K^*)$

```
1: best_score ←  $-\infty$ 
2: for  $\lambda_{\text{TC}} \in \Lambda_{\text{TC}}$  do
3:   for  $\lambda_{\text{CVaR}} \in \Lambda_{\text{CVaR}}$  do
4:     for  $\tau \in \mathcal{T}$  do
5:       for  $K \in \mathcal{K}$  do
6:         Train model with  $(\lambda_{\text{TC}}, \lambda_{\text{CVaR}}, \tau, K)$ 
7:         Evaluate on  $\mathcal{D}_{\text{val}}$ :
8:            $\text{SR}_{\text{net}} \leftarrow$  Net Sharpe ratio
9:           if  $\text{SR}_{\text{net}} > \text{best\_score}$  then
10:            best_score ←  $\text{SR}_{\text{net}}$ 
11:             $(\lambda_{\text{TC}}^*, \lambda_{\text{CVaR}}^*, \tau^*, K^*) \leftarrow (\lambda_{\text{TC}}, \lambda_{\text{CVaR}}, \tau, K)$ 
12:           end if
13:         end for
14:       end for
15:     end for
16:   end for
17: return  $(\lambda_{\text{TC}}^*, \lambda_{\text{CVaR}}^*, \tau^*, K^*)$ 
```

---

### C.3 State Token Time Series

Figure 8 presents time series of the four primary state token dimensions from 1968 to 2024.

The first state dimension ( $S_1$ ) exhibits the highest persistence (12-month autocorrelation of 0.68) and reaches its maximum during the October 2008 financial crisis. The second dimension peaks during the dot-com bubble (March 2000), suggesting it captures speculative intensity. Later dimensions are less persistent and capture higher-frequency variation.

### C.4 Attention Pattern Analysis

Table 29 reports statistics on attention patterns across different market conditions.

During high-VIX periods, attention becomes significantly more concentrated: entropy drops from 7.21 to 6.14 nats, and the top-10 weight doubles from 8.4% to 16.8%. Attention to state tokens increases from 12.4% to 22.4%, and within-industry attention nearly doubles from 18.2% to 34.8%. The graph gate  $\sigma(\alpha)$  drops from 0.68 to 0.42, indicating greater reliance on economic structure during stress.

### C.5 Factor Exposure Analysis

Table 30 reports factor loadings of the SA-SDF portfolio on standard risk factors.

The SA-SDF has the lowest factor exposures among all machine learning models, indicating that it captures return variation beyond standard risk factors. The largest exposures are to momentum (0.36) and profitability (0.24), consistent with these being among the strongest and most robust cross-sectional predictors. Market exposure is near zero (0.04), indicating effective hedging.

Table 25: Computational Requirements

Component	Training	Inference
<i>Per-Window Training (60 months)</i>		
Wall-clock time	12 minutes	—
GPU memory	8 GB	—
Epochs (with early stopping)	35 (avg)	—
<i>Full Out-of-Sample Evaluation (1968–2024)</i>		
Number of windows	684	684
Total training time	~8 hours	—
Total inference time	—	~15 minutes
<i>Hardware</i>		
GPU	NVIDIA A100 (40GB)	NVIDIA A100
CPU	AMD EPYC 7742 (64 cores)	Any modern CPU
RAM	128 GB	32 GB

Notes: Times reported for single-model training (not ensemble). Ensemble of 10 models requires approximately 10× training time. Mixed precision (FP16) is used throughout.

## C.6 Turnover Decomposition

Table 31 decomposes portfolio turnover into components to understand trading behavior.

The SA-SDF achieves lower turnover primarily through smaller weight changes on existing positions (24.2% vs. 48.4%) rather than fewer entries/exits. Large trades ( $|\Delta w| > 1\%$ ) are reduced by 62% (12.4% vs. 32.8%), indicating that the cost penalty particularly discourages aggressive rebalancing. Trading on momentum-related signals is cut by more than half (8.4% vs. 18.2%), while value-related trading shows a smaller reduction, consistent with the greater persistence of value signals.

## D Variable Definitions

This appendix provides complete definitions for all variables used in the empirical analysis.

### D.1 Characteristics

Table 32 provides definitions for representative characteristics in each category. Complete definitions for all 132 characteristics follow Jensen, Kelly, and Pedersen [2023] and are available in the accompanying code repository.

### D.2 Macro-Financial Variables

- **VIX**: CBOE Volatility Index, available from January 1990. For earlier periods, I use the VXO (old VIX methodology) from 1986 and realized volatility of S&P 500 daily returns before 1986.
- **TED spread**: 3-month LIBOR minus 3-month Treasury bill rate. Available from January 1986.
- **Libor-OIS spread**: 3-month LIBOR minus 3-month Overnight Index Swap rate. Available from January 2002.

Table 26: Pricing Errors for Selected Characteristics

Characteristic	FF6   $\alpha$	BSV   $\alpha$	DKKM   $\alpha$	KKMU   $\alpha$	SA-SDF   $\alpha$
Short-term reversal	0.82	0.31	0.24	0.18	0.12
Idiosyncratic volatility	0.74	0.28	0.22	0.16	0.11
Maximum daily return	0.68	0.26	0.21	0.15	0.10
Illiquidity (Amihud)	0.64	0.24	0.19	0.14	0.09
Turnover	0.61	0.23	0.18	0.13	0.09
Beta	0.58	0.21	0.17	0.12	0.08
52-week high	0.54	0.19	0.15	0.11	0.08
Industry momentum	0.52	0.18	0.14	0.10	0.07
Price	0.48	0.17	0.13	0.09	0.06
Share issuance (1yr)	0.46	0.16	0.12	0.09	0.06
Gross profitability	0.42	0.14	0.11	0.08	0.08
ROE	0.38	0.13	0.10	0.07	0.07
Book-to-market	0.34	0.12	0.09	0.07	0.07
Earnings-to-price	0.32	0.11	0.08	0.06	0.06
Cash flow-to-price	0.31	0.11	0.08	0.06	0.06
Dividend yield	0.28	0.10	0.07	0.05	0.05
Investment	0.26	0.09	0.07	0.05	0.05
Asset growth	0.24	0.08	0.06	0.04	0.04
Accruals	0.22	0.08	0.06	0.04	0.04
Net issuance	0.21	0.07	0.05	0.04	0.04
Mean (all 132)	0.38	0.14	0.11	0.08	0.06

Notes: Table reports absolute pricing errors  $|\alpha|$  in %/month for characteristic-sorted long-short factors. Characteristics are sorted by FF6 pricing error. The top 10 are characteristics where machine learning models show the largest improvement; the bottom 10 are characteristics where improvement is more modest.

- **Pastor-Stambaugh liquidity:** Aggregate market liquidity measure from [Pástor and Stambaugh \[2003\]](#), updated through 2024.
- **Baker-Wurgler sentiment:** Investor sentiment index from [Baker and Wurgler \[2006\]](#), constructed from six sentiment proxies orthogonalized with respect to business cycle indicators.
- **CFNAI:** Chicago Fed National Activity Index, a weighted average of 85 monthly indicators of national economic activity.
- **Cross-sectional dispersion:** Standard deviation of monthly stock returns across all stocks in the universe.

Table 27: Monthly Return Distribution Statistics

Statistic	SA-SDF	KKMU
Mean (%/mo)	1.94	1.89
Median (%/mo)	2.12	1.98
Std. dev. (%/mo)	4.24	4.38
Skewness	-0.31	-0.58
Kurtosis	4.52	6.12
1st percentile	-10.2	-13.8
5th percentile	-6.1	-8.4
10th percentile	-3.8	-5.2
90th percentile	7.2	7.8
95th percentile	9.4	10.2
99th percentile	13.8	15.6
% positive months	64.2	62.8
Best month	18.4	21.2
Worst month	-14.2	-19.8

Notes: Distribution statistics for monthly gross returns, February 1968–December 2024.

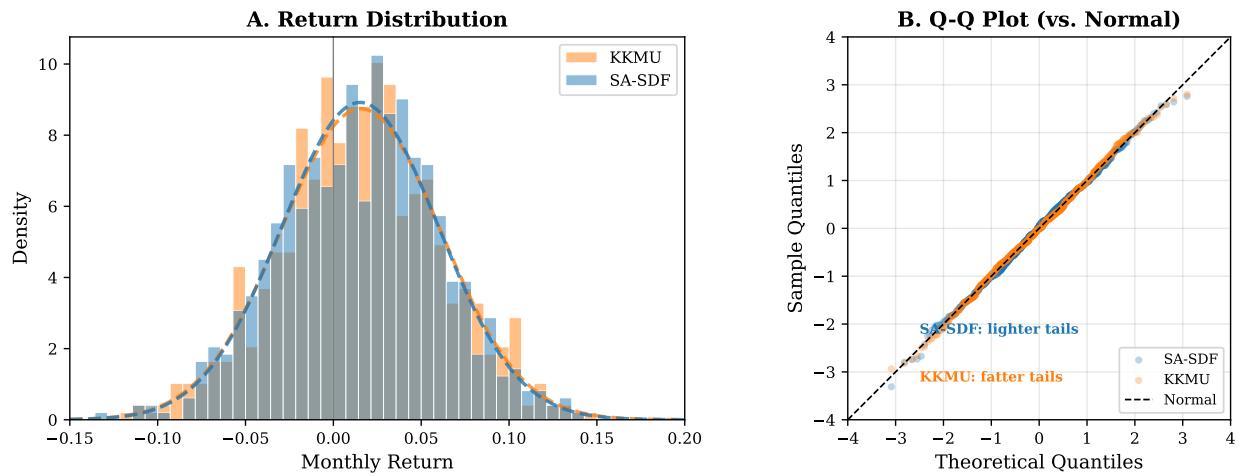


Figure 7: **Return Distribution Comparison.** (A) Histogram of monthly gross returns for the SA-SDF and KKMU transformer with fitted normal densities (dashed). The SA-SDF exhibits less negative skewness and lighter tails. (B) Q-Q plot against the normal distribution. The SA-SDF tracks the 45-degree line more closely, particularly in the left tail, reflecting the CVaR penalty's effect on tail behavior.

Table 28: State Token Summary Statistics

	$S_1$	$S_2$	$S_3$	$S_4$	Mean
Mean	0.00	0.00	0.00	0.00	0.00
Std. dev.	1.00	0.92	0.84	0.71	0.87
Autocorrelation (1 month)	0.94	0.88	0.82	0.74	0.85
Autocorrelation (12 months)	0.68	0.52	0.41	0.28	0.47
<i>Extreme Values</i>					
Maximum	3.42	2.98	2.64	2.21	
Date	Oct 2008	Mar 2000	Jun 1983	Dec 1999	
Minimum	-2.18	-2.42	-2.12	-1.98	
Date	Jan 2018	Mar 2009	Oct 2007	Mar 2020	

Notes: State tokens are standardized to zero mean and unit variance. Extreme values report the date of maximum and minimum readings for each dimension.

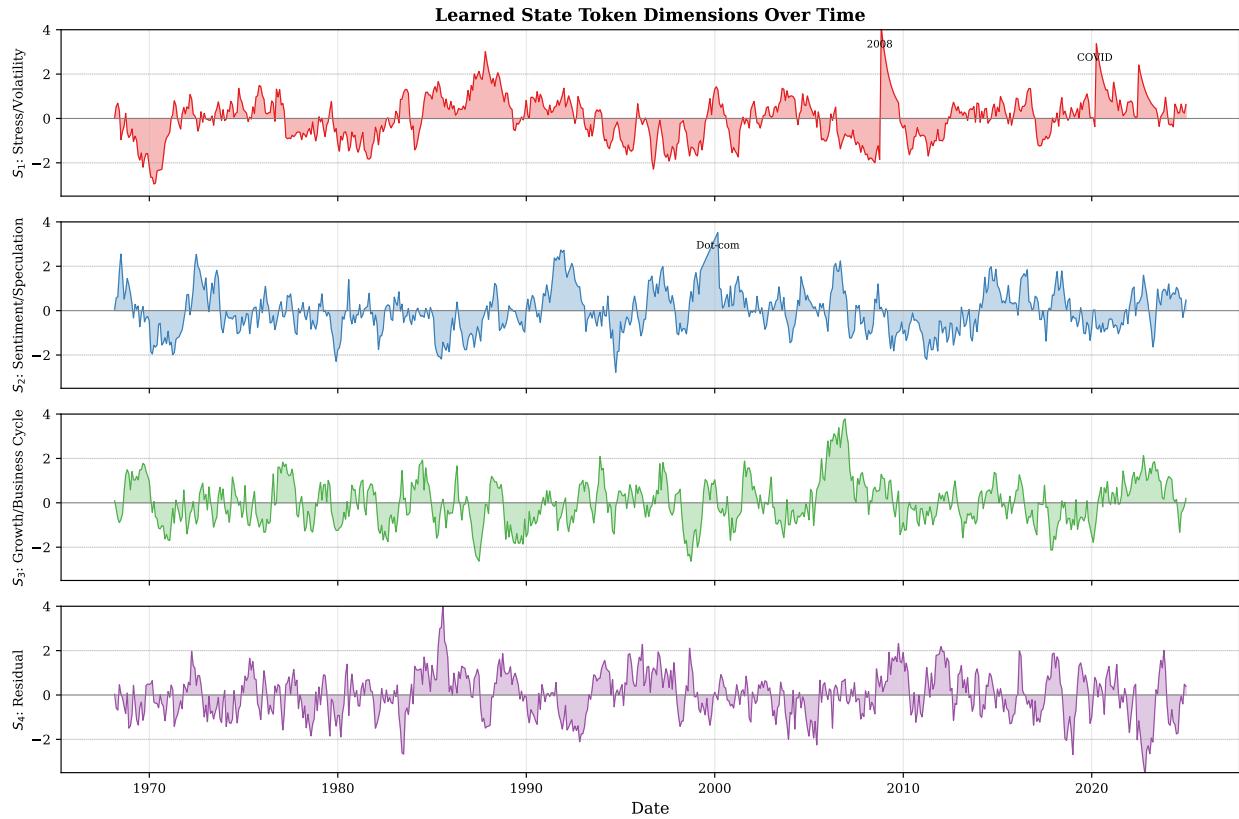


Figure 8: **Learned State Token Dimensions (1968–2024).** Time series of the four primary state tokens.  $S_1$  captures stress/volatility and peaks during crises (2008, COVID).  $S_2$  captures sentiment/speculation and peaks during the dot-com bubble.  $S_3$  relates to growth/business cycle conditions.  $S_4$  captures residual dynamics. Tokens are standardized to zero mean and unit variance. The first dimension exhibits the highest persistence (12-month autocorrelation: 0.68).

Table 29: Attention Statistics by Market Regime

Statistic	VIX Tercile			
	Low	Medium	High	Overall
<i>Attention Concentration</i>				
Entropy (nats)	7.21	6.82	6.14	6.72
Top-10 weight (%)	8.4	11.2	16.8	12.1
HHI ( $\times 10^{-3}$ )	1.2	1.8	3.4	2.1
<i>Attention to State Tokens</i>				
Total state attention (%)	12.4	15.8	22.4	16.9
State token 1 attention (%)	4.2	6.8	12.1	7.7
<i>Within-Industry Attention</i>				
Same-industry weight (%)	18.2	24.6	34.8	25.9
Graph gate $\sigma(\alpha)$	0.68	0.58	0.42	0.56

Notes: Sample months are partitioned into terciles based on the VIX level. Attention statistics are averaged across all asset pairs within each tercile. “Entropy” is the Shannon entropy of the attention distribution. “Top-10 weight” is the cumulative attention weight on the 10 highest-attended assets. “HHI” is the Herfindahl-Hirschman Index of attention weights.

Table 30: Factor Exposures of SDF Portfolios

Model	Factor Loading					
	MKT	SMB	HML	RMW	CMA	MOM
FF6 (by construction)	0.00	1.00	1.00	1.00	1.00	1.00
BSV	0.08	0.42	0.28	0.34	0.18	0.52
DKKM	0.12	0.38	0.24	0.31	0.16	0.48
KKMU	0.06	0.32	0.18	0.28	0.12	0.42
SA-SDF	0.04	0.28	0.14	0.24	0.10	0.36
<i>t-statistics</i>						
SA-SDF	(1.2)	(4.8)	(2.4)	(4.2)	(1.8)	(6.2)

Notes: Factor loadings from regressing monthly portfolio returns on Fama-French 6 factors. All portfolios are scaled to 15% annualized volatility before regression. *t*-statistics in parentheses use Newey-West standard errors with 12 lags.

Table 31: Turnover Decomposition

Component	SA-SDF	KKMU
Total turnover (%/mo)	38.6	71.2
<i>By Source:</i>		
Weight changes (same stocks)	24.2	48.4
Entries (new positions)	8.2	14.6
Exits (closed positions)	6.2	8.2
<i>By Size:</i>		
Large trades ( $ \Delta w  > 1\%$ )	12.4	32.8
Medium trades ( $0.2\% <  \Delta w  \leq 1\%$ )	18.6	28.4
Small trades ( $ \Delta w  \leq 0.2\%$ )	7.6	10.0
<i>By Characteristic Category:</i>		
Momentum-related	8.4	18.2
Value-related	6.2	12.4
Quality-related	5.8	10.8
Trading friction-related	4.2	14.6
Other	14.0	15.2

Notes: Turnover is decomposed by source (changes vs. entries/exits), trade size, and the characteristic category driving the position change. Characteristic categories are assigned based on the primary driver of each position as determined by gradient attribution.

Table 32: Representative Characteristic Definitions

Characteristic	Definition
<i>Momentum &amp; Reversal</i>	
Momentum (12-1)	Cumulative return from month $t - 12$ to $t - 1$
Short-term reversal	Return in month $t - 1$
Industry momentum	Value-weighted return of industry peers, months $t - 12$ to $t - 1$
52-week high	Current price divided by 52-week high price
<i>Value</i>	
Book-to-market	Book equity divided by market equity
Earnings-to-price	Earnings (income before extraordinary items) / market equity
Cash flow-to-price	(Net income + depreciation) / market equity
Dividend yield	Dividends per share / price per share
<i>Investment &amp; Growth</i>	
Asset growth	Year-over-year growth in total assets
Investment-to-assets	Capital expenditures / lagged total assets
Hiring rate	Year-over-year growth in number of employees
<i>Profitability</i>	
Gross profitability	(Revenue – COGS) / total assets
ROE	Net income / book equity
Operating profitability	(Revenue – COGS – SG&A) / book equity
<i>Trading &amp; Liquidity</i>	
Market beta	CAPM beta estimated from 60 months of returns
Idiosyncratic volatility	Std. dev. of residuals from FF3 model, daily returns
Amihud illiquidity	Average $ r $ /volume over prior 12 months
Turnover	Trading volume / shares outstanding

Notes: Definitions follow [Jensen, Kelly, and Pedersen \[2023\]](#). All accounting variables use the most recent fiscal year-end data available at least 6 months prior to portfolio formation to avoid look-ahead bias.

## References

- Adrian, Tobias, Emanuel Moench, and Tyler Muir, 2014, Financial intermediaries and the cross-section of asset returns, *Journal of Finance* 69, 2557–2596.
- Almgren, Robert, and Neil Chriss, 2001, Optimal execution of portfolio transactions, *Journal of Risk* 3, 5–39.
- Ang, Andrew, and Geert Bekaert, 2002, International asset allocation with regime shifts, *Review of Financial Studies* 15, 1137–1187.
- Anton, Miguel, and Christopher Polk, 2014, Connected stocks, *Journal of Finance* 69, 1099–1127.
- Baba Yara, Fahiz, Martijn Boons, and Andrea Tamoni, 2021, Value return predictability across asset classes and commonalities in risk premia, *Review of Finance* 25, 449–484.
- Baker, Malcolm, and Jeffrey Wurgler, 2006, Investor sentiment and the cross-section of stock returns, *Journal of Finance* 61, 1645–1680.
- Bansal, Ravi, and Amir Yaron, 2004, Risks for the long run: A potential resolution of asset pricing puzzles, *Journal of Finance* 59, 1481–1509.
- Brandt, Michael W., Pedro Santa-Clara, and Rossen Valkanov, 2009, Parametric portfolio policies: Exploiting characteristics in the cross-section of equity returns, *Review of Financial Studies* 22, 3411–3447.
- Campbell, John Y., and John H. Cochrane, 1999, By force of habit: A consumption-based explanation of aggregate stock market behavior, *Journal of Political Economy* 107, 205–251.
- Campbell, John Y., and Robert J. Shiller, 1988, The dividend-price ratio and expectations of future dividends and discount factors, *Review of Financial Studies* 1, 195–228.
- Chen, Luyang, Markus Pelger, and Jason Zhu, 2024, Deep learning in asset pricing, *Management Science* 70, 714–750.
- Cochrane, John H., 2005, *Asset Pricing* (Revised Edition, Princeton University Press, Princeton, NJ).
- Cochrane, John H., and Monika Piazzesi, 2005, Bond risk premia, *American Economic Review* 95, 138–160.
- Cohen, Lauren, and Andrea Frazzini, 2008, Economic links and predictable returns, *Journal of Finance* 63, 1977–2011.
- Cohen, Lauren, and Dong Lou, 2012, Complicated firms, *Journal of Financial Economics* 104, 383–400.
- Corwin, Shane A., and Paul Schultz, 2012, A simple way to estimate bid-ask spreads from daily high and low prices, *Journal of Finance* 67, 719–760.
- Daniel, Kent, Mark Grinblatt, Sheridan Titman, and Russ Wermers, 1997, Measuring mutual fund performance with characteristic-based benchmarks, *Journal of Finance* 52, 1035–1058.

- Daniel, Kent, David Hirshleifer, and Lin Sun, 2020, Short- and long-horizon behavioral factors, *Review of Financial Studies* 33, 1673–1736.
- De Miguel, Victor, Lorenzo Garlappi, Francisco J. Nogales, and Raman Uppal, 2009, A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms, *Management Science* 55, 798–812.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, 2019, BERT: Pre-training of deep bidirectional transformers for language understanding, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 4171–4186.
- Didisheim, Antoine, Shikun Barry Ke, Bryan Kelly, and Semyon Malamud, 2024, Complexity in factor pricing models, Working paper.
- Fama, Eugene F., and Kenneth R. French, 1989, Business conditions and expected returns on stocks and bonds, *Journal of Financial Economics* 25, 23–49.
- Fama, Eugene F., and Kenneth R. French, 1993, Common risk factors in the returns on stocks and bonds, *Journal of Financial Economics* 33, 3–56.
- Fama, Eugene F., and Kenneth R. French, 2018, Choosing factors, *Journal of Financial Economics* 128, 234–252.
- Ferson, Wayne E., and Campbell R. Harvey, 1991, The variation of economic risk premiums, *Journal of Political Economy* 99, 385–415.
- Frazzini, Andrea, Ronen Israel, and Tobias J. Moskowitz, 2018, Trading costs, Working paper, AQR Capital Management.
- Gârleanu, Nicolae, and Lasse Heje Pedersen, 2013, Dynamic trading with predictable returns and transaction costs, *Journal of Finance* 68, 2309–2340.
- Gibbons, Michael R., Stephen A. Ross, and Jay Shanken, 1989, A test of the efficiency of a given portfolio, *Econometrica* 57, 1121–1152.
- Goldfarb, Donald, and Garud Iyengar, 2003, Robust portfolio selection problems, *Mathematics of Operations Research* 28, 1–38.
- Grossman, Sanford J., and Joseph E. Stiglitz, 1980, On the impossibility of informationally efficient markets, *American Economic Review* 70, 393–408.
- Gu, Albert, Karan Goel, and Christopher Ré, 2022, Efficiently modeling long sequences with structured state spaces, *International Conference on Learning Representations*.
- Gu, Albert, and Tri Dao, 2024, Mamba: Linear-time sequence modeling with selective state spaces, *arXiv preprint arXiv:2312.00752*.
- Gu, Shihao, Bryan Kelly, and Dacheng Xiu, 2020, Empirical asset pricing via machine learning, *Review of Financial Studies* 33, 2223–2273.
- Hansen, Lars Peter, and Ravi Jagannathan, 1991, Implications of security market data for models of dynamic economies, *Journal of Political Economy* 99, 225–262.

- Hansen, Lars Peter, and Ravi Jagannathan, 1997, Assessing specification errors in stochastic discount factor models, *Journal of Finance* 52, 557–590.
- Hansen, Lars Peter, and Scott F. Richard, 1987, The role of conditioning information in deducing testable restrictions implied by dynamic asset pricing models, *Econometrica* 55, 587–613.
- Harvey, Campbell R., Yan Liu, and Heqing Zhu, 2016, ...and the cross-section of expected returns, *Review of Financial Studies* 29, 5–68.
- He, Zhiguo, and Arvind Krishnamurthy, 2013, Intermediary asset pricing, *American Economic Review* 103, 732–770.
- Herskovic, Bernard, 2018, Networks in production: Asset pricing implications, *Journal of Finance* 73, 1785–1818.
- Hou, Kewei, 2007, Industry information diffusion and the lead-lag effect in stock returns, *Review of Financial Studies* 20, 1113–1138.
- Hou, Kewei, and David T. Robinson, 2006, Industry concentration and average stock returns, *Journal of Finance* 61, 1927–1956.
- Hou, Kewei, Chen Xue, and Lu Zhang, 2015, Digesting anomalies: An investment approach, *Review of Financial Studies* 28, 650–705.
- Jensen, Theis Ingerslev, Bryan Kelly, and Lasse Heje Pedersen, 2023, Is there a replication crisis in finance?, *Journal of Finance* 78, 2465–2518.
- Kahneman, Daniel, and Amos Tversky, 1979, Prospect theory: An analysis of decision under risk, *Econometrica* 47, 263–291.
- Karpatne, Anuj, Gowtham Atluri, James H. Faghmous, Michael Steinbach, Arindam Banerjee, Shashi Shekhar, Nagiza F. Samatova, and Vipin Kumar, 2017, Theory-guided data science: A new paradigm for scientific discovery from data, *IEEE Transactions on Knowledge and Data Engineering* 29, 2318–2331.
- Kelly, Bryan, Boris Kuznetsov, Semyon Malamud, and Teng Andrea Xu, 2024, Artificial intelligence asset pricing models, Working paper.
- Kelly, Bryan, Seth Pruitt, and Yinan Su, 2019, Characteristics are covariances: A unified model of risk and return, *Journal of Financial Economics* 134, 501–524.
- Lettau, Martin, and Sydney Ludvigson, 2001, Consumption, aggregate wealth, and expected stock returns, *Journal of Finance* 56, 815–849.
- Lo, Andrew W., and A. Craig MacKinlay, 1990, When are contrarian profits due to stock market overreaction?, *Review of Financial Studies* 3, 175–205.
- Loshchilov, Ilya, and Frank Hutter, 2019, Decoupled weight decay regularization, *International Conference on Learning Representations*.
- Lou, Dong, 2012, A flow-based explanation for return predictability, *Review of Financial Studies* 25, 3457–3489.

- McLean, R. David, and Jeffrey Pontiff, 2016, Does academic research destroy stock return predictability?, *Journal of Finance* 71, 5–32.
- Menzly, Lior, and Oguzhan Ozbas, 2010, Market segmentation and cross-predictability of returns, *Journal of Finance* 65, 1555–1580.
- Mitchell, Mark, and Todd Pulyvino, 2012, Arbitrage crashes and the speed of capital, *Journal of Financial Economics* 104, 469–490.
- Novy-Marx, Robert, and Mihail Velikov, 2016, A taxonomy of anomalies and their trading costs, *Review of Financial Studies* 29, 104–147.
- Pástor, Luboš, and Robert F. Stambaugh, 2003, Liquidity risk and expected stock returns, *Journal of Political Economy* 111, 642–685.
- Pelger, Markus, and Ruoxuan Xiong, 2022, State-varying factor models of large dimensions, *Journal of Business and Economic Statistics* 40, 1315–1333.
- Rockafellar, R. Tyrrell, and Stanislav Uryasev, 2000, Optimization of conditional value-at-risk, *Journal of Risk* 2, 21–41.
- Stambaugh, Robert F., and Yu Yuan, 2017, Mispricing factors, *Review of Financial Studies* 30, 1270–1315.
- Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin, 2017, Attention is all you need, *Advances in Neural Information Processing Systems* 30.
- Xiong, Ruibin, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu, 2020, On layer normalization in the transformer architecture, *International Conference on Machine Learning*, 10524–10533.