Learning Interpretable Time-Inhomogeneous Markov Operators for Financial Time Series

Jan Rovirosa[1] **Jesse Schmolze**[1]
[1]University of Wisconsin – Madison, Madison, WI, USA

Abstract

Modeling the dynamics of non-stationary stochastic systems requires balancing the representational power of deep learning with the mathematical transparency of classical models. While classical Markov transition operators provide explicit, theoretically grounded rules for system evolution, their empirical estimation collapses due to severe data sparsity when applied to high-resolution, high-noise environments. We explore this statistical barrier using financial time series as a canonical, real-world testbed. To overcome the degeneracy of empirical counting, we introduce a framework that utilizes neural networks strictly as parameterization engines to generate explicit, time-varying Markov transition matrices. By constraining the neural network to output its predictions as a formal stochastic operator, we maintain complete structural interpretability. We demonstrate that these learned operators successfully capture complex regime shifts, which can be quantified using established operator-theoretic metrics such as row heterogeneity and the Dobrushin coefficient. Furthermore, rather than enforcing the Chapman–Kolmogorov equations as a rigid structural requirement, we repurpose them as a localized diagnostic tool to pinpoint specific temporal windows where first-order memory assumptions break down. Ultimately, this framework demonstrates how neural networks can be constrained to make rigorous, classical operator analysis viable for complex real-world time series.

1 Introduction

Learning the dynamics of non-stationary stochastic systems is a central challenge in machine learning. We are often caught between two extremes: interpretable probabilistic models (like Markov chains), which offer clear structural insights but struggle to adapt to constantly evolving environments, and modern neural sequence models, which handle complex dependencies with ease but hide their logic inside black-box latent representations [1, 11]. This paper proposes a middle ground: a framework that uses the flexibility of neural networks to model the analytical structures of classical stochastic processes in real-time.

Financial returns provide a high-stakes setting for this challenge. Market dynamics are noisy, heavy-tailed, and subject to abrupt regime changes [9]. A standard way to study these shifts is to discretize returns into a finite state space $\mathcal{S} = \{s_1, \dots, s_n\}$ and model the transitions using a matrix $A \in [0, 1]^{n\times n}$ [2, 8]. However, a conflict arises when we seek high resolution. To capture the nuance of heavy-tailed returns, we require a fine discretization (a large n). But as the number of bins increases, the number of possible transitions grows quadratically (n^2). In the finite, noisy window of financial data, most of these n^2 transitions are never observed. This produces degenerate operators; matrices filled with zeros and high-variance noise that fail to represent the true underlying dynamics and become useless for actual analysis.

We address this degeneracy by replacing counting with neural parameterization. Instead of tabulating how many times s_i moved to s_j, we train a neural network to learn the underlying mapping from current features F_t and states X_t to a probability distribution over the next state. Mathematically, we model the system as a feature-conditioned, time-inhomogeneous Markov chain. At each time t, the dynamics are governed by a transition operator $\widehat{A}_t$ constructed row-by-row:

$$\widehat{A}_t(i, \cdot) = \hat{p}_\theta(\cdot \mid X_t = s_i, F_t) \quad (1)$$

By evaluating the network for every possible $s_i \in \mathcal{S}$, we reconstruct a full, smooth transition matrix for that specific moment in time. This approach transforms transition estimation into a conditional density estimation problem [4]. The neural network creates the framework, filling in the sparse entries of the matrix by interpolating patterns across similar states and features.

The primary advantage of this framework is that it yields an object that is both flexible and mathematically interpretable. Because the output is a standard stochastic operator, we can apply classical diagnostic tools that are unavailable to generic deep learning models [11]: we can plot transition heatmaps to see how "rules" shift during market crashes vs. calm periods, we can measure row heterogeneity (how much the current state actually matters) and dispersion (how uncertain the market is), and we can use the Chapman–Kolmogorov equation to check if the model's long-term forecasts are logically consistent with its short-term steps, providing a sanity check for the model's internal logic.

Contributions:

- We introduce a neural parameterization for time-inhomogeneous Markov operators that addresses the sparsity problem of fine discretization while remaining fully interpretable.
- We demonstrate that these learned operators are more stable and less degenerate than empirical estimators in high-noise financial settings.
- We provide a suite of operator-level diagnostics, including multi-step consistency checks and regime-window analysis, to evaluate the structural integrity of the learned dynamics.
- We show through ablation that explicit Markov conditioning is a critical inductive bias for stable learning and temporal generalization for several configurations.

2 Related Work

2.1 Classical Markov structure in economics and finance

Markov structure has long been attractive in economics and finance because it provides an explicit probabilistic description of regime change. Hamilton's seminal Markov-switching framework [1] showed that nonstationary macroeconomic dynamics can be modeled through regime-dependent evolution governed by a discrete-state Markov law, making shifts between economic phases analytically interpretable. In credit-risk settings, transition matrices play a similarly central role: Lando and Skodeberg [2] study rating migration using continuous-time observations and show that transition estimation itself, together with departures from simple Markov assumptions, is a core empirical issue. More broadly, financial returns are known to exhibit heavy tails, volatility clustering, and other stylized facts [9], which motivate regime-sensitive probabilistic models while also making fine-grained empirical estimation noisy and statistically fragile.

2.2 Time-varying and input-conditioned Markov dynamics

A natural extension of classical Markov modeling is to allow the transition law to vary with time or observable covariates. Mettle et al. [8] analyze exchange rates using time-inhomogeneous finite-state Markov chains, illustrating the value of allowing transition structure to evolve across market conditions rather than assuming a single stationary operator. In machine learning, Bengio and Frasconi's input–output HMM [5] provides an important precursor: latent-state transitions are conditioned on an observed input sequence, linking classical Markov structure with learned, input-dependent dynamics.

Our setting is different in two important respects. First, the state of interest is not latent: it is an explicit discretization of the observed return process. Second, the primary object we wish to estimate and inspect is the induced transition operator itself. This distinction matters because our goal is not only sequential prediction, but also structural analysis of how transition rules vary over time.

2.3 Neural probabilistic modeling without explicit operators

From an estimation perspective, our task is a conditional density estimation problem over a finite state space. Under linear parameterizations, this reduces to standard multinomial probabilistic models, for which convex optimization provides the classical theoretical and algorithmic framework [3]. Our contribution is to replace that linear parameterization with a neural one, while preserving the same explicit probabilistic target:

$$\mathbb{P}(X_{t+h} \mid X_t, F_t) \quad \text{or} \quad \mathbb{P}(Y_t^{(h)} \mid X_t, F_t). \tag{2}$$

This connects our work to broader neural approaches to probabilistic prediction. Mixture density networks [4] model full conditional output distributions rather than point estimates, and modern forecasting architectures such as DeepAR [6] learn time-varying predictive distributions with autoregressive neural sequence models. Surveys such as [11] show that this probabilistic viewpoint is now standard in deep time-series forecasting. However, these approaches typically represent dynamics through latent recurrent states or distribution parameters at the observation level; they do not produce an explicit stochastic operator on a fixed state space that can be directly inspected and analyzed as a transition law.

2.4 Neural–Markov hybrids and the position of our framework

Several works combine Markov ideas with neural architectures, but for different purposes. Awiszus and Rosenhahn [7] construct neural networks capable of simulating Markov-chain behavior in stochastic generative settings. Bellemare et al. [10], in distributional reinforcement learning, likewise emphasize that discrete probability distributions can be treated as primary predictive objects rather than reduced to expectations. These perspectives are broadly compatible with our emphasis on learning distributions directly.

The central difference is that our framework is operator-first. We do not use a neural network to replace classical stochastic structure; we use it strictly as a parameterization engine for that structure. At each time t, the network generates a full, explicit, feature-conditioned transition operator whose rows remain valid probability distributions over a fixed discretized state space. This is precisely what allows classical operator analysis to survive in a setting where direct empirical counting collapses. Because the learned object is an actual stochastic operator rather than an opaque latent representation, we can inspect it through transition heatmaps, quantify state dependence through row heterogeneity and related contraction metrics, and use Chapman–Kolmogorov composition as a localized diagnostic of where first-order Markov closure is most and least plausible.

In this sense, our contribution is not simply a neural forecasting model with Markov flavor. It is a framework for making classical time-inhomogeneous Markov analysis viable in high-resolution, high-noise financial settings where empirical transition estimation becomes too sparse to be useful.

3 Problem Formulation

Our objective is to model the evolution of a stochastic system whose transition rules change over time. We represent the system as a feature-conditioned, time-inhomogeneous Markov model and focus on the induced transition operators.

3.1 State Space and Discretization

Let $\{P_t\}_{t\geq 0}$ denote adjusted daily closing prices and define the one-day return

$$r_t := \frac{P_t - P_{t-1}}{P_{t-1}}. \tag{3}$$

To enable a structural analysis of regime dynamics, we discretize returns into a finite state space $\mathcal{S} = \{s_1, \ldots, s_n\}$ via a map $\phi : \mathbb{R} \to \mathcal{S}$, and define the Markov state

$$X_t := \phi(r_t) \in \mathcal{S}. \tag{4}$$

We prioritize relatively fine discretizations (large n). While this provides a detailed view of the return distribution, it introduces a sparsity problem: the number of potential state-to-state transitions scales as n^2, which can exceed the available sample support, making empirical (count-based) transition estimators unreliable.

3.2 Feature-Conditioned One-Step Dynamics

Let $F_t \in \mathbb{R}^d$ denote observable covariates available at time t (exogenous indicators and/or lagged features). We assume a first-order Markov property conditional on F_t:

$$\begin{aligned}\mathbb{P}(X_{t+1} = s_j \mid X_t = s_i,\ X_{t-1}, X_{t-2}, \ldots,\ F_t) = \\ =\ \mathbb{P}(X_{t+1} = s_j \mid X_t = s_i,\ F_t).\end{aligned} \tag{5}$$

Unlike stationary Markov chains, the transition law is allowed to vary with time through its dependence on F_t [5, 8].

3.3 Two Operator Views and Prediction Targets

Our experiments use two complementary label constructions, leading to two operator interpretations.

(i) Forward-return (distributional) targets. For horizon $h \geq 1$, define the forward return from $t+1$ to $t+1+h$ by

$$R_t^{(h)} := \frac{P_{t+1+h} - P_{t+1}}{P_{t+1}}, \tag{6}$$

and discretize it into m_h bins to obtain $Y_t^{(h)} \in \mathcal{Y}^{(h)}$ with $|\mathcal{Y}^{(h)}| = m_h$. This yields a (generally rectangular) population operator $A_t^{(h)} \in [0,1]^{n \times m_h}$ defined by

$$A_t^{(h)}(i,j) := \mathbb{P}(Y_t^{(h)} = j \mid X_t = s_i,\ F_t). \tag{7}$$

(ii) State-to-state targets (operator diagnostics). For operator diagnostics (including Chapman–Kolmogorov checks), we instead set

$$Y_t^{(h)} := X_{t+h} \in \mathcal{S}, \tag{8}$$

which produces a square population operator $A_t^{(h)} \in [0,1]^{n \times n}$:

$$A_t^{(h)}(i,j) := \mathbb{P}(X_{t+h} = s_j \mid X_t = s_i,\ F_t). \tag{9}$$

In this setting, multi-step composition is well-defined on the common state space $\mathcal{S}$.

3.4 Neural Parameterization: a Row-wise Operator Estimator

To avoid unstable counting in high-dimensional discretizations, we estimate transition rows using a neural model $f_\theta^{(h)}$. For a given state s_i and features F_t, the model outputs a probability vector over the label space:

$$\widehat{A}_t^{(h)}(i,\cdot) := f_\theta^{(h)}(s_i, F_t) \in \Delta^{m_h - 1}, \tag{10}$$

where Δ^{m_h-1} is the probability simplex (and $m_h = n$ in the state-to-state diagnostic setting). To construct the full estimated operator at time t, we fix F_t and evaluate the network for each $s_i \in \mathcal{S}$:

$$\text{for } i = 1, \ldots, n: \quad \widehat{A}_t^{(h)}(i,\cdot) \leftarrow f_\theta^{(h)}(s_i, F_t). \tag{11}$$

This replaces sparse, noisy transition counts with a shared-parameter estimator that produces well-defined operator rows even when particular transitions are rarely observed. In other words, for a fixed feature snapshot F_t, we can explicitly construct the full operator $\widehat{A}_t^{(h)} \in [0,1]^{n \times m_h}$ by evaluating the same network once per input state s_i. This yields a concrete matrix of transition probabilities at each time t (square when $m_h = n$), rather than a latent representation.

4 Methods and Models

4.1 Data, Alignment, and Preprocessing

Our empirical study focuses on a single equity (JPM) together with a collection of exogenous covariates obtained from public and commercial financial data sources. The covariates comprise (i) daily market and macroeconomic indicators (e.g., policy-rate and credit-spread proxies) and (ii) lower-frequency firm fundamentals (e.g., quarterly accounting variables and ratios). Because many features are reported at lower frequency than prices, all covariates are aligned to the daily grid by carrying forward the most recently observed value until the next release date; short internal gaps are filled by linear interpolation, while features requiring extensive extrapolation are excluded.

Each feature is standardized using training-set statistics only, producing a normalized feature vector $F_t \in \mathbb{R}^d$. When using reduced feature sets, we rank features by mutual information with the training labels and retain the top-k features.

4.2 Discretization Choices

We discretize one-day returns into $n = 55$ states using quantile binning fit on the training segment. For horizon-h forward-return labels, we discretize $R_t^{(h)}$ into m_h bins (typically $m_h \in \{10, 20, 35, 55\}$). For operator diagnostics we set $Y_t^{(h)} = X_{t+h}$, which yields square operators on $\mathcal{S}$.

4.3 Neural Operator Parameterization

For a given horizon h, the model inputs the concatenation of a one-hot state encoding and the feature vector,

$$z_t \ := \ [\, e(X_t)\,;\, F_t\,] \in \mathbb{R}^{n+d}, \tag{12}$$

and produces logits in $\mathbb{R}^{m_h}$ through an MLP $g_\theta^{(h)}$. The predicted distribution is

$$\hat{p}_\theta^{(h)}(\cdot \mid X_t, F_t) \ = \ \mathrm{softmax}\Big(g_\theta^{(h)}(z_t)\Big) \in \Delta^{m_h - 1}. \tag{13}$$

This defines an estimated operator row $\widehat{A}_t^{(h)}(X_t, \cdot)$, and when $m_h = n$ it yields an interpretable stochastic transition matrix.

4.4 Optimization and Regularization

Models are trained using Adam with early stopping based on validation negative log-likelihood. To regularize learning under noisy discretization, we optionally replace the one-hot label with a smoothed target distribution over neighboring bins. Let $Y_t^{(h)} \in \{1, \ldots, m_h\}$ denote the discrete label at horizon h, and let $\tilde{\mathbf{q}}_t^{(h)} \in \Delta^{m_h - 1}$ denote its smoothed target distribution (e.g., a Gaussian kernel over bin indices with bandwidth $\sigma^{(h)}$). Given predicted probabilities

$$\hat{\mathbf{p}}_t^{(h)} := \hat{p}_\theta^{(h)}(\cdot \mid X_t, F_t) \in \Delta^{m_h - 1},$$

we minimize the smoothed cross-entropy objective

$$\mathcal{L}_{\mathrm{smooth}}^{(h)}(\theta) \ = \ -\sum_{t \in \mathcal{T}_{\mathrm{train}}} \sum_{j=1}^{m_h} \tilde{q}_{t,j}^{(h)} \log \hat{\mathbf{p}}_t^{(h)}(j). \tag{14}$$

All preprocessing steps (bin-edge fitting, feature standardization, feature ranking) are performed using training data only. Reported results use chronological train/validation/test splits rather than random i.i.d. splits.

4.5 State-Free Ablation Baseline

To isolate the contribution of explicit Markov state conditioning, we also consider a state-free baseline that removes $e(X_t)$ from the input:

$$\hat{p}_{\theta,\mathrm{sf}}^{(h)}(\cdot \mid F_t) = \mathrm{softmax}\Big(g_{\theta,\mathrm{sf}}^{(h)}(F_t)\Big). \tag{15}$$

In the state-to-state setting ($m_h = n$), this baseline corresponds to an operator whose rows are identical at each time t, serving as a reference for diagnosing the extent to which learned dynamics depend on the Markov state.

5 Experiments

To validate our neural parameterization framework, we design an experimental suite that explicitly tests the limits of classical transition estimation and evaluates the structural interpretability of the learned operators.

5.1 Dataset, Horizons, and Splits

We evaluate the framework on a single-equity case study (JPM). The dataset consists of T daily trading observations, each with an aligned feature vector $F_t \in \mathbb{R}^d$. To strictly avoid look-ahead bias, all evaluations utilize chronological splits. For each look-ahead horizon $h \in \{1, 2, 5, 10\}$, we construct a horizon-specific dataset and apply a fixed train/validation/test split of $70\%/15\%/15\%$ along the time axis.

The Markov state space is fixed to a fine discretization of $n = 55$ bins ($X_t \in \mathcal{S}$), fit via quantiles on the training segment to ensure approximately balanced marginal state visitation.

5.2 Prediction Targets

As formulated in Section 3, we operationalize two complementary label constructions:

1. **Forward-return targets:** The forward return $R_t^{(h)}$ is discretized into $m_h \in \{10, 20, 35, 55\}$ bins. This yields rectangular operators $A_t^{(h)} \in [0,1]^{n \times m_h}$ and is used to assess general transition dynamics under varying label resolutions.

2. **State-to-state targets:** For internal structural diagnostics, we set $Y_t^{(h)} := X_{t+h} \in \mathcal{S}$. This yields square operators $A_t^{(h)} \in [0,1]^{n \times n}$, allowing operators to be composed over time on a shared state space.

5.3 Models and Baselines

All neural models share an identical base Multi-Layer Perceptron (MLP) architecture and differ solely in their inductive biases regarding the Markov state.

State-Conditioned Transition Model. Our primary model, which parameterizes the operator row $\widehat{A}_t^{(h)}(X_t, \cdot)$ via $[e(X_t); F_t]$. By explicitly conditioning on the current state X_t, this model acts as a dynamic operator generator.

State-Free Ablation Baseline. To isolate the value of the Markov inductive bias, we train a baseline conditioned only on features ($\hat{p}_{\theta,\mathrm{sf}}^{(h)}(\cdot \mid F_t)$). By definition, this model outputs an operator with identical rows at each time step (row-invariant). It serves as a negative control: if the state-conditioned model collapses to look like this baseline, the Markov state carries no useful transition information.

Count-Based Estimators. To represent the classical approach, we evaluate empirical estimators derived from training counts of $(X_t, Y_t^{(h)})$. These include a marginal baseline, a direct conditional estimator, and a backoff-smoothed estimator (with validation-tuned hyperparameters). These baselines are included specifically to demonstrate the degeneracy of empirical counting under fine discretization.

5.4 Training Protocol

The shared neural architecture is an MLP with hidden widths $64 \to 128 \to 256 \to 128 \to 64$, using GELU activations and dropout ($p = 0.2$). Models are optimized via Adam with weight decay and gradient clipping. Early stopping is governed by validation negative log-likelihood. To account for the noise inherent in discretized returns, we test both hard cross-entropy and a smoothed-label objective that allocates partial mass to neighboring bins.

5.5 Evaluation Metrics (Operator-Focused)

In alignment with our framework, our evaluation prioritizes operator structure and diagnostic interpretability over raw point prediction.

Degeneracy and Sparsity. To quantify the failure modes of classical methods, we measure the sparsity of the empirical count matrices $C \in \mathbb{N}^{n \times m_h}$ on the training data. Metrics include the fraction of absolute zero cells and the fraction of cells falling below a minimum observation threshold (e.g., < 5 samples).

Operator Interpretability Diagnostics. For the square one-step operators $\widehat{A}_t^{(1)}$, we compute time-varying structural statistics:

- *Row Heterogeneity:* The average pairwise total-variation (TV) distance between rows, acting as a direct, mathematical measure of state dependence.
- *Row Entropy:* The mean entropy across rows, measuring the operator's forward dispersion.
- *Dobrushin Coefficient:* A measure of the operator's contractive properties (row separation in ℓ_1).

Chapman–Kolmogorov (CK) Consistency. In the state-to-state setting, we measure the divergence between the directly estimated h-step operator $\widehat{A}_t^{(h)}$ and the composed product $\widehat{A}_t^{(1:h)}$. This is quantified using row-wise KL divergence and TV distance, localized over time to identify regime windows where the first-order Markov assumption weakens.

Sanity Checks. Finally, to ensure the learned operators reflect valid probability distributions, we compute the test-set Negative Log-Likelihood (NLL) relative to a marginal baseline. We also assess distributional reliability via probability integral transform (PIT) histograms and event-level metrics (Expected Calibration Error and Brier score) for coarse regimes like negative returns. We assess robustness through multi-seed repeats and block bootstrapping over time to ensure conclusions are not artifacts of initialization.

6 Results

6.1 The Sparsity Barrier: Why Empirical Estimation Fails

To demonstrate the necessity of neural parameterization, we first expose the severe limitations of classical count-based Markov estimation in high-noise, high-resolution settings. For each configuration (h, m_h) in the forward-return setting, we construct the empirical count matrix $C^{(h)} \in \mathbb{N}^{n \times m_h}$ on the training split:

$$C^{(h)}(i, j) := \#\{t \in \mathcal{T}_{\mathrm{train}} : X_t = s_i,\ Y_t^{(h)} = j\}. \quad (16)$$

Although the marginal states X_t are defined via quantile binning (ensuring roughly equal visitation overall), the *joint* transition structure becomes catastrophically sparse once we condition on the current state.

As shown in Figure 1 and Figure 2, the empirical degeneracy increases rapidly with label resolution. For instance, at a fine label discretization of $m_h = N = 55$ bins, nearly 100% of the transition cells contain fewer than 5 observations, even at a 1-day horizon ($h = 1$). This formalizes the core problem: direct empirical counting yields a degenerate, high-variance matrix that is essentially empty, rendering it useless for studying market dynamics. This statistical collapse is the primary motivation for replacing direct counting with a regularized, feature-conditioned operator estimator.

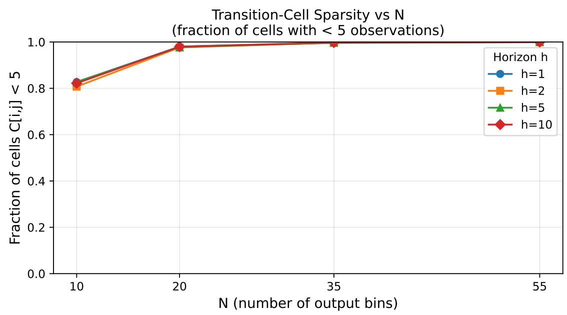

Figure 1: Empirical degeneracy increases rapidly with label resolution. Even with quantile-based discretization, the empirical count matrix $C^{(h)}$ becomes dominated by low-count entries as m_h increases, formalizing why direct counting yields unstable transition estimates in fine-grained operator estimation.

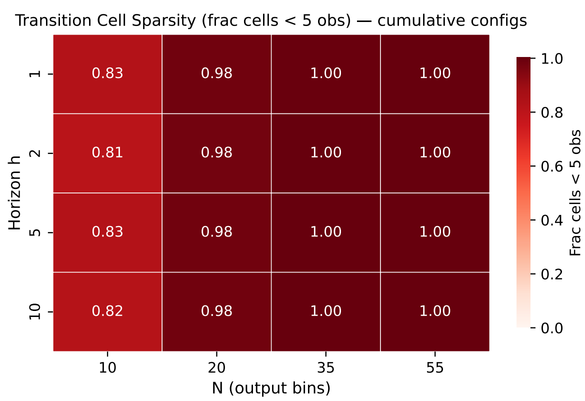

Figure 2: Empirical transition degeneracy for the forward-return tasks. Each cell reports the fraction of entries in the training-set count matrix $C^{(h)}$ with fewer than 5 observations. Values at 1 (such as the $N = 35, 55$ columns) indicate that empirical counting is completely dominated by sparse cells.

6.2 Visualizing Market Rules: Inspectable Transition Operators

The immediate payoff of our neural parameterization is the ability to generate a complete, mathematically sound transition operator $\widehat{A}_t^{(1)} \in [0, 1]^{n \times n}$ for any given time t by evaluating the network for all n possible input states under the same feature snapshot F_t.

Our model's outputs can be directly inspected as probability heatmaps (Figure 3). To ensure fair comparisons, all snapshots share a global color scale ($v_{\min} = 0$ and a single global $v_{\max}$ computed across the snapshots).

The visual contrast between the architectures highlights the value of the Markov inductive bias. The state-conditioned model produces complex, diagonal-heavy structures whose rows vary significantly, capturing the persistence and shifting dispersion of the market based on its current state. In contrast, the state-free baseline produces row-invariant operators (horizontal stripes); by ignoring the Markov state X_t, it is forced to output the exact same next-step distribution regardless of where the market currently is, effectively throwing away the transition structure.

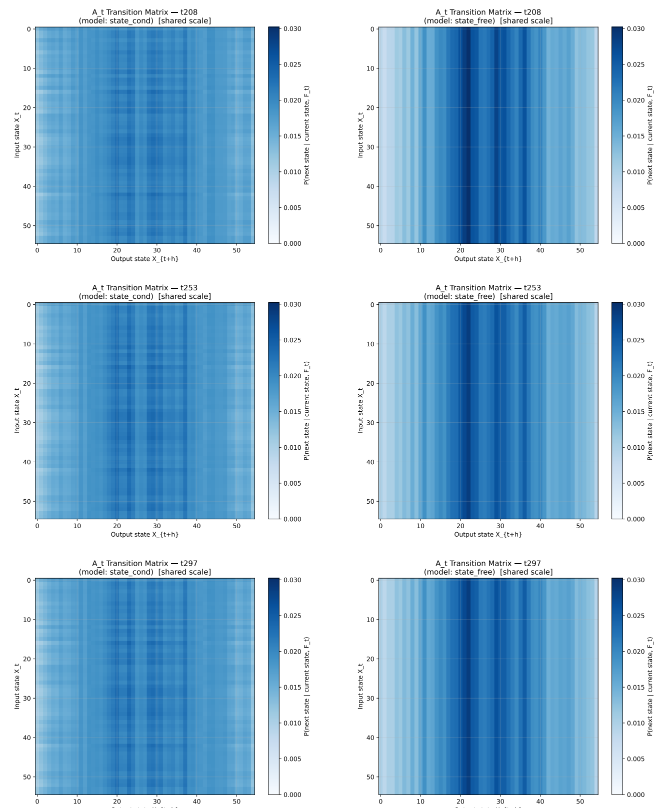

Figure 3: Heatmap snapshots of the learned one-step operator $\widehat{A}_t^{(1)}$ at three selected regime indices (rows: $t \in \{208, 253, 297\}$). Left column: state-conditioned model, displaying rich transition dynamics. Right column: state-free baseline, which is row-invariant at each t by construction.

6.3 Quantitative Operator Diagnostics and Regime Shifts

Visualizing operators provides qualitative intuition, but their true scientific value lies in our ability to subject them to quantitative, operator-theoretic diagnostics. We compute three key statistics from $\widehat{A}_t^{(1)}$ over the full time series:

1. **Row Heterogeneity** (average pairwise TV distance between rows): This is a direct mathematical measure of state dependence. If all rows are identical (heterogeneity is zero), the current Markov state is irrelevant to the system's future.

2. **Row Entropy**: A measure of the operator's dispersion, reflecting market uncertainty.

3. **Dobrushin Coefficient**: A measure of the operator's contractive properties and mixing time.

Figure 4 tracks these metrics over time. The state-conditioned model exhibits strong, dynamic row heterogeneity that spikes and drops in response to shifting market regimes. The state-free baseline, serving as a sanity check, collapses to exactly zero heterogeneity by design. This confirms that our learned operators successfully extract interpretable, time-varying structural shifts in the market's underlying dynamics.

Figure 4: Regime diagnostics computed from the learned one-step operator $\widehat{A}_t^{(1)}$ over the full time series. The state-conditioned model exhibits nontrivial time variation and identifiable windows where operator structure changes. The state-free baseline serves as a sanity check, with row heterogeneity collapsing toward zero.

6.4 Testing Structural Consistency: The Chapman–Kolmogorov Diagnostic

Because our model outputs standard stochastic operators on a common state space, we can evaluate its internal structural logic using the time-inhomogeneous Chapman–Kolmogorov (CK) equations. We compare the model's direct prediction of an h-step transition ($\widehat{A}_t^{(h)}$) against the mathematical composition of its 1-step transitions over the same window:

$$\widehat{A}_t^{(1:h)} := \widehat{A}_t^{(1)} \widehat{A}_{t+1}^{(1)} \cdots \widehat{A}_{t+h-1}^{(1)}. \tag{17}$$

Figure 5 reports the time-resolved CK discrepancy (mean KL divergence). Crucially, we do not frame a perfectly zero CK error as a mandatory empirical requirement. In real financial data under fine discretization, a strict first-order Markov closure in X_t is only an approximation. Instead, we utilize CK as a structural diagnostic tool: periods where the CK discrepancy spikes indicate specific temporal windows where the first-order Markov assumption breaks down. This type of deep logical diagnostic localizes where the operator view is most and least internally consistent, an analysis entirely unavailable to generic "black-box" sequence models [11].

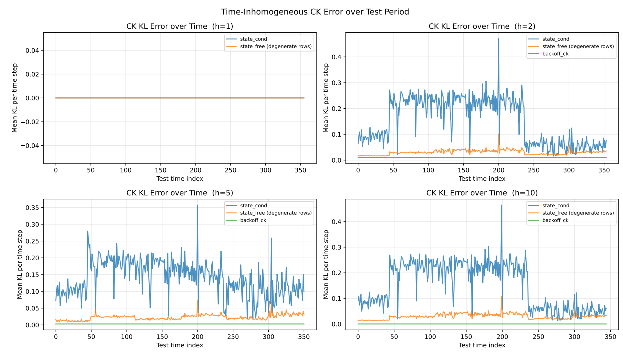

Figure 5: Time-resolved CK KL discrepancy over the test period for varying horizons h. The CK error is not uniform in time; its spikes align with windows where operator diagnostics indicate structural regime changes. We also include a smoothed count-based estimator (in green) as a classical baseline reference.

6.5 Sanity Checks

We also conducted probabilistic sanity checks to ensure the learned operators are empirically grounded. Specifically, we evaluated the model's predictive log-likelihood relative to a marginal baseline (Δ NLL). For several configurations ($h = 1$ and $N = 35$, $h = 10$ and $N = 20$, and $h = 10$ and $N = 55$), the model achieves a positive Δ NLL, demonstrating that the state-conditioned operators extract valid predictive signal beyond random marginal frequencies.

7 Discussion

7.1 Bridging Deep Learning and Structural Finance

The central premise of this work is that deep learning in finance does not have to come at the cost of structural interpretability. By utilizing neural networks strictly as parameterization engines for explicit Markov transition operators, we overcome the sparsity and degeneracy barriers that limit classical empirical estimation under fine discretizations. The resulting state-conditioned operators ($\widehat{A}_t^{(h)}$) provide a rigorously bounded probability space that can be visually inspected and mathematically diagnosed. Rather than relying on opaque latent embeddings typical of many modern sequence models [11], our framework allows practitioners to extract time-varying market rules (such as shifting dispersion, persistence, and state-dependence) using established metrics like row heterogeneity and the Dobrushin coefficient.

7.2 Rethinking the Markov Assumption via CK Diagnostics

A key conceptual shift in our framework is our use of the Chapman–Kolmogorov (CK) equation as a dynamic diagnostic tool. In real-world financial time series, the first-order Markov assumption (where X_{t+1} depends only on X_t and F_t) is inherently an approximation. Our time-resolved CK analysis (Section 6.4) reveals that this assumption does not fail uniformly; rather, it breaks down during specific, identifiable market regimes. Spikes in CK divergence serve as valuable structural signals indicating periods where the market exhibits extended memory or where unobserved latent factors dominate the transition dynamics. This type of localized, logical debugging is entirely absent in generic sequence models [11].

7.3 Data limitations and Future Work

The primary objective of this work was to establish a mathematically rigorous framework for neural operator parameterization and to validate its structural interpretability. While the predictive gains (Δ NLL) are moderate in absolute terms, they represent a substantial achievement when contextualized by the severe constraints of the empirical setting. Our study was deliberately restricted to a single equity and relied heavily on fundamental features that are often artificially interpolated or carried forward over quarterly or yearly intervals. Extracting a valid, structured predictive signal from such a low signal-to-noise environment confirms the robustness of the explicit Markov formulation and proves that the framework operates as intended.

Building on this foundational proof-of-concept, our future work will focus on scaling the model to maximize predictive accuracy and deepening our analytical toolset:

- **Data Scaling and Cross-Asset Dynamics:** To drastically improve predictive accuracy, we intend to scale the neural parameterization to ingest high-frequency, synchronous data across a vast multi-asset universe. Expanding from a single-asset state space to portfolio-level operators will allow the model to leverage cross-sectional correlation dynamics, directly informing downstream applications like dynamic asset allocation and systematic risk management.

- **Higher-Order Markov Formulations:** The current architecture assumes a first-order Markov closure. Future iterations will test higher-order state conditioning (e.g., explicitly incorporating X_{t-1}, X_{t-2}) to provide the model with richer historical context. This will allow the network to better capture momentum and extended market memory, yielding a more precise transition operator without sacrificing the explicit, inspectable nature of the state space.

- **Deepening Operator Diagnostics:** We plan to expand the suite of structural diagnostics applied to the learned operators. Key directions include analyzing Chapman–Kolmogorov consistency explicitly partitioned by regime windows, evaluating the stability of the transition operator $\widehat{A}_t$ under feature perturbations, and formally tracking mixing times and contractivity measures as market conditions evolve.

Ultimately, by restricting a highly flexible neural network to output a mathematically coherent structural form, this framework provides a transparent, well-calibrated tool for studying non-stationary market dynamics. Because it yields explicit, readable transition probabilities rather than opaque latent vectors [11], the learned operators can be directly integrated into downstream financial applications, ranging from systematic regime identification and market stress warning systems, to dynamic portfolio optimization and tactical asset allocation. It demonstrates that the analytical rigor of classical stochastic processes and the representational power of modern deep learning can be successfully unified to solve complex, high-stakes problems.

8 Conclusion

This paper introduced a framework for learning time-varying Markov transition operators in settings where classical empirical estimation becomes unreliable. In high-resolution financial discretizations, direct count-based estimators rapidly collapse under sparsity: most possible transitions are rarely observed, producing degenerate and high-variance matrices that no longer support meaningful structural analysis. Our response is to replace empirical tabulation with neural parameterization, using a shared-parameter network to estimate conditional transition rows while preserving the output as an explicit stochastic operator.

The primary value of this approach is interpretability at the level of the operator. By constraining the model to output formal transition operators, we retain access to classical operator-level analysis in a noisy, non-stationary environment. The learned operators can be inspected directly through heatmaps, compared across market conditions, and summarized through interpretable statistics such as row heterogeneity, row entropy, and the Dobrushin coefficient. In this way, the framework occupies a middle ground between rigid classical Markov models and opaque neural sequence models: it uses neural networks to make operator estimation feasible while preserving the explicit probabilistic structure that makes Markov models analytically useful.

Empirically, our results support this positioning. The neural estimator produces dense, well-defined transition operators across time. The state-conditioned model exhibits meaningful time variation and nontrivial state dependence, whereas the state-free ablation collapses to row-invariant behavior by construction. This confirms that conditioning on the current discretized state is necessary to obtain genuinely state-dependent operator dynamics, and in several configurations it also improves held-out likelihood.

A second contribution is methodological. Rather than treating the Chapman–Kolmogorov equations as a hard structural condition that real data must satisfy exactly, we use them as a diagnostic lens. The observed CK discrepancies are informative precisely because they vary over time: they localize windows where a first-order Markov approximation is more or less adequate, and where additional memory or latent structure may be needed. This reframing turns a classical consistency relation into a practical tool for local structural debugging of learned dynamics.

Taken together, these results show that neural networks can be used not as black-box replacements for stochastic models, but as disciplined parameterization engines for them. This makes classical Markov operator analysis viable in a financial regime where empirical counting fails. More broadly, it suggests a general strategy for modeling complex non-stationary systems: preserve the explicit mathematical object of interest, and use modern function approximators only to estimate it more effectively.

References

[1] Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57(2), 357–384.

[2] Lando, D., & Skodeberg, T. M. (2002). Analyzing rating transitions and rating drift with continuous observations. *Journal of Banking & Finance*, 26(2), 423–444.

[3] Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

[4] Bishop, C. M. (1994). Mixture density networks. Technical Report NCRG/94/004, Neural Computing Research Group, Aston University.

[5] Bengio, Y., & Frasconi, P. (1995). An input output HMM architecture. *Advances in Neural Information Processing Systems*, 7, 427–434.

[6] Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3), 1181–1191.

[7] Awiszus, M., & Rosenhahn, B. (2018). Markov chain neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

[8] Mettle, F. O., Boateng, L. P., Quaye, E. N. B., Aidoo, E. K., & Seidu, I. (2022). Analysis of exchange rates as time-inhomogeneous Markov chain with finite states. *Journal of Probability and Statistics*, 2022.

[9] Cont, R. (2001). Empirical properties of asset returns: stylized facts and statistical issues. *Quantitative Finance*, 1(2), 223–236.

[10] Bellemare, M. G., Dabney, W., & Munos, R. (2017). A distributional perspective on reinforcement learning. *International Conference on Machine Learning (ICML)*, 449–458.

[11] Lim, B., & Zohren, S. (2021). Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194).