

The Clock of Regimes: Matrix-Analytic Solutions for Continuous-Time Survival in Discrete HMMs using the R KRONX Package*

Oscar Linares^{1*}  and Ričards Bulavas² 

¹OIS Market Research Group, Alberta iela 1-7, Rīga, LV-1050 Latvia

²University of Latvia, Raiņa bulvāris 19, Rīga, LV-1586 Latvia

*Corresponding Author

Oscar Linares, OIS Market Research Group, Alberta iela 1-7, Rīga, LV-1050 Latvia.

Submitted: 2026, May 16; **Accepted:** 2026, May 26;

Published: 2026, May 29

Citation: Linares, O., Bulavas, R. (2026). The Clock of Regimes: Matrix-Analytic Solutions for Continuous-Time Survival in Discrete HMMs using the R KRONX Package*. *J Math Techniques Comput Math*, 5(4), 1-47.

Abstract:

Hidden Markov Models (HMMs) are widely used to identify latent regime dynamics in financial time series, yet standard R implementations report state membership probabilities rather than the structural geometry of residence and absorption (ruin). This paper introduces **KRONX**, an R package that extends discrete-time HMMs into a survival-analytic framework through a matrix-operator chain. Starting from a Student-t HMM transition matrix A , the package constructs a hazard-adjusted sub-stochastic operator Q , a sub-generator $K = Q - I$, and the fundamental matrix $N = -K^{-1}$ of cumulative expected sojourn times before absorption—adapting the mean-residence-time machinery from compartmental analysis to financial regime dynamics. The core analytical contribution is an exact, closed-form derivation of residence weights and a finite-horizon ruin bound, avoiding the numerical noise and $O(S \cdot T)$ cost of Monte Carlo simulation. An empirical workflow using E-mini S&P 500 futures demonstrates the full pipeline from model specification through residency geometry and risk quantification. Written in pure R with no external dependencies beyond **stats** and **utils**, **KRONX** is designed for auditability, reproducibility, and integration into existing HMM-based analysis workflows.

Keywords: Matrix-Analytic Methods, Hidden Markov Model, Regime Switching Models, Survival, Hazard, Residence Time, KRONX R Package.

1 Introduction

Regime-switching models serve as a foundational framework for capturing the time-varying dynamics of asset returns. The standard approach—pioneered by Hamilton (1989) and adapted for financial applications by Rydén et al. (1998)—characterizes log returns as realizations from a K -component mixture model, where transition probabilities are governed by a latent Markov chain. While several R packages implement this architecture—notably **depmixS4** (Visser and Speekenbrink, 2010), **HMM** (Himmelman, 2010), **MSwM** (Sánchez-Espigares and López-Moreno, 2021), and the financially-focused **fHMM** (Oelschläger et al., 2023)—their primary utility lies in state estimation and parameter inference. Consequently, while these tools identify the current regime of a time series, they offer little insight into the duration of state residency or the system’s proximity to structural absorption and ruin.

This paper introduces the **KRONX R** package which fills that gap. The package takes the estimated transition matrix from a fitted Hidden Markov Model (HMM) and converts it into a survival-analytic operator system that characterizes the residence geometry and finite-horizon fragility of the Markov chain. The intellectual foundation draws from compartmental analysis, and compartmental systems mean residence times (Berman and Schoenfeld, 1956; Hearon, 1972, 1981; Eisenfeld, 1981; Covell et al., 1984; Linares et al., 1987, 1988; Jacquez and Simon, 1993; Jacquez, 2002). The central object in the **KRONX R** package is a continuous-time generator and its associated resolvent operator, $N = -K^{-1}$, which measures expected cumulative time spent across regimes. This operator-based formulation permits a structural decomposition of persistence beyond one-step transition probabilities and connects regime dynamics to survival analysis through state-dependent hazard rates (Kleinbaum and Klein, 2012; Gao and He, 2025). By integrating hazard rates, the model allows for state-dependent exit and connects regime dynamics to survival analysis and tail risk.

This paper is structured as follows. Section 2 provides the necessary background and a review of the relevant literature. Subsection 2.1 formalizes the open system dynamics of the COR model, while Subsection 2.2 characterizes its fundamental matrix geometry. The transition to a survival analysis framework is developed in Subsubsection 2.2.1. Subsection 2.3 is devoted to the directional change (DC) framework and its alignment with the COR framework. Subsubsection 2.3.1 discusses how DC may be viewed as an event-time observation layer feeding the COR model. Subsubsection 2.3.1 discusses DC thresholds and how they may behave once persistence accumulates. Subsubsection 2.3.3 discusses how DC-derived indicators may be integrated with the COR framework. Subsubsection 2.3.4 discusses the synthesis of DC, HMM, and COR models. Subsection 2.4 discusses

the COR framework’s move beyond the limitations of Gaussian assumptions. Subsection 2.5 discusses the COR framework’s inherent Bayesian interpretation.

2 Background and Literature Review

In Hamilton-style regime switching:

$$S_t \in \{1, \dots, K\}$$

represents the hidden regime process. The clock of regimes (COR) model accepts this hidden regime structure, but argues that transition probabilities alone are insufficient to describe systemic fragility.

Classical HMMs focus on:

$$P(S_t = j \mid S_{t-1} = i) = a_{ij}$$

while COR asks how long does risk survive inside a regime architecture once hazard and persistence interact? This leads to the COR operator chain:

$$A \rightarrow Q \rightarrow K \rightarrow N$$

where:

- A = HMM transition matrix,
- $Q = \text{diag}(1 - \epsilon)A$ = open survival operator,
- $K = Q - I$ = sub-generator,
- $N = -K^{-1}$ = survival/residence operator.

Hence, K represents the dynamics of the system conditional on survival (Table 1).

Feature	Classical HMM	COR Framework
Primary Objective	Detects regimes	Measures survival geometry
Dynamics	Measures switching	Measures persistence under hazard
Analytical Focus	Transition-centric	Duration-centric
Core Output	Hidden state probabilities	Structural fragility operators

Table 1: Comparative analysis of HMM and COR model, highlighting the shift from transition-centric classification to duration-centric structural analysis.

Hamilton’s (Hamilton, 1989, 2016) key insight was that economies oscillate between recession and expansion regimes. The COR model generalizes this idea by introducing the following:

2.1 Open-System Dynamics

Hamilton assumes a closed Markov chain:

$$\sum_j a_{ij} = 1.$$

The COR model introduces leakage:

$$Q = \text{diag}(1 - \epsilon)A,$$

meaning:

- Probability mass can leave the system;
- Regimes are mortal: the probability mass of remaining in the current state decays toward zero over time as $t \rightarrow \infty$;
- Persistence possesses a formal survival structure.

This is the defining conceptual innovation of the model.

2.2 Fundamental Matrix Geometry

Hamilton's framework focuses on the retrospective and real-time identification of market states through the lens of smoothed and filtered probabilities. Filtered probabilities provide an optimal estimate of the current regime based on all information available up to the present time, whereas smoothed probabilities utilize the entire sample period to offer a more precise, backward-looking assessment of historical regime shifts. Central to this analysis is the concept of transition persistence, which quantifies the likelihood of the system remaining within a specific state. By examining these probabilities, Hamilton characterizes the *stickiness* of economic regimes, allowing for a rigorous distinction between temporary market fluctuations and fundamental structural changes in the underlying data-generating process.

In contrast, the COR framework shifts the analytical focus toward the fundamental survival dynamics of the system by examining the survival/residence operator, $N = -K^{-1}$. Rather than focusing on instantaneous transitions, this approach evaluates the cumulative expected residence time, providing a rigorous measure of the total duration a process is expected to remain within a specific state. By analyzing the integrated survival geometry of the regime, the framework captures the structural *shape* of persistence, effectively quantifying how long the system endures before reaching a boundary. Ultimately, this characterizes persistence under hazard, treating regimes not

as permanent loops, but as decaying open systems whose persistence is determined by the rate of leakage toward a boundary.

Equivalently:

$$N = \int_0^{\infty} e^{Kt} dt$$

This transforms regime switching into a survival analysis problem.

2.2.1 The Transformation to Survival Analysis

In a classical Markov framework, e^{Kt} is the transition semigroup, representing the probability of remaining in a specific state configuration at time t . By integrating this probability from 0 to ∞ , we are essentially summing up all the moments of survival the system experiences.

By defining $N = \int_0^{\infty} e^{Kt} dt$, we are performing a Laplace transform (evaluated at $s = 0$) on the transition dynamics. This results in several key insights of the COR framework:

- **From Frequency to Duration:** While Hamilton-style HMMs analyze the frequency of state transitions via the A matrix, this integral calculates the area under the survival curve. This shifts the focus from the probability of an instantaneous switch to the total mass of time accumulated within a specific regime.
- **The Operator Inverse:** For a stable, non-singular sub-generator K , the integral evaluates directly to $-K^{-1}$. This identity establishes the inverse of the sub-generator as the formal mathematical equivalent of total accumulated persistence.
- **Survival Geometry:** The term e^{Kt} characterizes the trajectory of stochastic decay. Integrating this term captures the entire “geometry” of regime resistance; in highly stable systems, the rate of decay is attenuated, resulting in a larger integral and, consequently, a longer expected residence time.

2.3 Directional Change (DC) and COR

The Directional Change (DC) framework (Tsang, 2010; Aloud et al., 2011; Tsang, 2017; Glattfelder et al., 2011; Tsang et al., 2017; Bakhach et al., 2018; Tsang and Chen, 2018) fundamentally reimagines the temporal basis of financial analysis by abandoning fixed clock time in favor of an event-based sampling methodology. Rather than observing the market at arbitrary intervals—such as daily or hourly closes—DC samples data only when the price moves by a predefined threshold, effectively filtering out stochastic noise to focus exclusively on structural turning points. This approach is deeply aligned with COR philosophy, as both methodologies reject the linear passage of time as a primary metric. By prioritizing the *event* over the *interval*, the DC framework treats market

movements as a sequence of meaningful state changes, providing a natural empirical substrate for the survival/residence operators used to measure regime persistence and structural fragility.

2.3.1 Event Time vs Clock Time

The COR framework aligns seamlessly with the Directional Change (DC) approach because it replaces the static observation of price with a dynamic analysis of state endurance. By focusing on the structural properties of a regime rather than just its classification, COR provides a robust mathematical language for event-based markets.

Traditional HMMs are tied to

$$t = 1, 2, 3, \dots$$

fixed increments. But COR already behaves like an event-driven survival system because

$$N = -K^{-1}$$

integrates expected occupancy across persistence geometry. This means DC can be viewed as an event-time observation layer feeding a survival-operator regime model.

2.3.2 DC Thresholds and COR Hazard

The DC event definition is given by

$$\left| \frac{P_t - P_{EXT}}{P_{EXT}} \right| \geq \theta$$

This threshold mechanism is conceptually analogous to COR hazards.

In COR,

$$\epsilon_i = \epsilon_{\min} + \frac{C_{\text{hazard}}}{\nu_i},$$

where

- fat tails imply larger hazard,
- unstable states leak probability mass faster.

Mechanism	Directional Change (DC)	COR Framework
Trigger	Threshold crossing	Hazard activation
Validation	Event confirmation	Structural instability
Criticality	Extreme points	Regime stress states
Extension	Overshoots	Persistence cascades

Table 2: Mapping of Directional Change events to COR structural operators.

DC detects market discontinuities, while COR measures how dangerous those discontinuities become once persistence accumulates.

2.3.3 DC Indicators as COR State Variables

The integration of Directional Change (DC) indicators as state variables provides a robust empirical foundation for the COR framework. Metrics such as Total Movement (TMV), which captures the price magnitude of a directional event, and T (Time to Complete Trend), which measures the duration required to reach a specific threshold, act as direct proxies for the physical work performed by a market regime. When coupled with R (Time-Adjusted Return), these indicators serve as extremely natural inputs for the survival/residence operator N . Rather than relying on arbitrary clock-time snapshots, COR can utilize these DC-derived variables to calibrate the persistence geometry of the system. This allows the framework to map the relationship between price-action intensity and structural stability, effectively quantifying the rate at which market regimes accumulate *fatigue* or *stress* as they move toward an inevitable state of absorption.

DC Indicator	DC Definition	COR Interpretation
TMV	Total Movement	State shock magnitude
T	Time to complete trend	Residence duration
R	Time-adjusted return	Hazard-adjusted velocity

Table 3: Translation of Directional Change indicators into COR structural variables.

The parameters displayed in Table 3 suggest a full hybrid system

$$X_t = (TMV_t, T_t, R_t)$$

feeding

$$P(X_t | S_t = i)$$

inside a Student- t HMM/COR structure. This presents a major research opportunity.

Contextual Significance

- **TMV → State Shock Magnitude:** Within the COR framework, Total Movement (TMV) quantifies the cumulative displacement across the persistence geometry. A high TMV indicates a significant injection of “shock” or volatility into the active regime prior to its eventual absorption.
- **T → Residence Duration:** This serves as the empirical realization of the expected residence time N . While T captures the observed duration of a specific directional event, N provides the formal probabilistic expectation of that duration derived from the sub-generator K .

- **R → Hazard-Adjusted Velocity:** By normalizing returns relative to event duration, R captures the “velocity” at which the system traverses its state space. In this context, it represents the rate of probability mass leakage toward structural termination under the current hazard profile.

2.3.4 From Transitions to Trajectories: The HMM-DC-COR Synthesis

The synthesis of Directional Change (DC) indicators and Hidden Markov Models (HMM) provides a powerful mechanism for real-time market characterization, particularly when integrated via Bayesian updating. In this architecture, DC indicators like TMV and T act as the primary *sensors*, filtering price action into discrete, event-driven signals that inform the HMM’s state transition logic. As new market data arrives, Bayesian updating allows for the continuous refinement of regime probabilities, ensuring that the model’s beliefs remain synchronized with shifting volatility profiles. COR extends this architecture naturally by transforming these updated state probabilities into structural metrics. Rather than stopping at state identification, COR utilizes the Bayesian-refined inputs to calibrate the sub-generator K , effectively moving beyond “what regime are we in” to a formal analysis of how much life remains in the current state before structural exhaustion or ruin ensues.

The progression becomes (Table 4):

Framework	Main Goal
Time-series HMM	Detect hidden regimes
DC + HMM	Detect event-driven regime shifts
COR	Measure fragility and survival geometry

Table 4: Evolution of regime-switching methodologies from detection to structural survival analysis.

COR effectively adds:

A. Survival Analysis

$$N = -K^{-1}$$

B. Spectral Fragility

$$\rho(N) = \frac{1}{\min |\lambda(K)|}$$

C. Ruin Geometry

$$P(\text{ruin}) \leq 1 - e^{-\bar{\lambda}qT}$$

D. Residence-Time Dynamics

$$\pi_{\text{res}} \propto \pi_{qs}^T N$$

None of these objects exist in the classical HMM/DC framework.

2.4 Gaussian HMM Limitation vs COR Student-t Architecture

While the current section assumes Gaussian emissions, the COR framework offers a significant advancement by addressing the structural underestimation of crisis persistence and tail fragility inherent in classical regime-switching models. By moving beyond the limitations of Gaussian assumptions, COR integrates Student- t emissions to more accurately reflect the empirical distribution of asset returns. This foundation enables fat-tail hazard coupling, which accounts for the accelerated likelihood of state transitions during periods of extreme volatility. The architecture is further strengthened by tail-dependent leakage, allowing the model to treat structural exhaustion as a direct function of market stress. By shifting the focus toward these survival dynamics, the COR framework evolves regime analysis from a classification task into a rigorous study of structural survival geometry, providing a more resilient representation of how market states endure or collapse under pressure.

2.5 Naïve Bayes and COR Bayesian Updating

The COR framework inherently supports a natural Bayesian interpretation by transforming standard regime-switching outputs into dynamic structural assessments. Within this architecture, posterior state probabilities—derived from the filtered and smoothed estimates of the underlying Markov chain—serve as the foundation for characterizing the current market environment. These probabilities are then mapped into a posterior hazard structure, which quantifies the instantaneous risk of state termination based on observed market evidence. Ultimately, this allows for the derivation of a posterior ruin geometry, where the survival operator $N = -K^{-1}$ is continuously updated to reflect the most likely expected residence time and structural fragility of the current regime. By integrating these Bayesian layers, the COR framework ensures that the measured persistence of a system is always conditioned on the most recent empirical shocks.

The stored Bayesian update structure:

$$P(H | D) = \frac{P(D | H)P(H)}{P(D | H)P(H) + P(D | \neg H)(1 - P(H))}, \quad (1)$$

fits naturally into the COR framework, which integrates these components into a unified analytical workflow that moves beyond static state detection.

Eq. 1 shows Bayes' Theorem expressed in a specific form often used for binary classification or diagnostic testing. In plain English, it describes how to update a belief in a hypothesis (H) after new data (D) is observed. Here is the breakdown: The formula calculates the probability that a hypothesis is true given that we have observed new evidence.

- The Left Side: $P(H | D)$ is the **Posterior Probability**. It represents the new, updated

confidence in the hypothesis after observing the data.

- The Top Right (Numerator): This is the probability of the data occurring if the hypothesis were true, multiplied by the original baseline probability, $P(H)$, that the hypothesis was true to begin with.
- The Bottom Right (Denominator): This represents the total probability of observing the data, $P(D)$, under all possible circumstances. It combines:
 - The probability of observing the data if the hypothesis is true.
 - The probability of observing the data if the hypothesis is not true, denoted by $(\neg H)$.

By leveraging dynamic hazard updating, the system continuously calibrates the sub-generator K as new market events arrive, ensuring that the modeled risk of transition remains synchronized with real-time volatility. This feeds directly into a rolling fragility estimation, where the survival operator N is recalculated over a moving window to capture the evolving structural strength of the regime. Consequently, the framework enables posterior ruin probability tracking, providing a rigorous, evidence-based measure of the likelihood that the current market configuration will succumb to structural absorption (ruin). This continuous feedback loop transforms the model into a responsive monitoring system capable of quantifying regime exhaustion as it develops.

2.6 The Principal Conceptual Departure

The principal distinction between these analytical perspectives lies in their fundamental inquiry: while Classical Regime Models focus on a diagnostic classification by asking, “Which regime are we in?,” the COR model shifts the focus to a prognostic, structural evaluation, asking, “How structurally survivable is the system while inside that regime architecture?” This represents a completely different ontological question, moving the model’s objective from simple state identification to the assessment of internal resilience (Table 5). By making this transition, the COR model effectively converts the task of regime detection into an analysis of survival geometry under structural fragility, treating the market not as a series of labels, but as a bounded system whose endurance is defined by its resistance to absorption.

Table 5: Conceptual comparison between standard HMMs and the KRONX framework.

Feature	Standard HMM Approach	KRONX (Ontological Shift)
Focus	Membership (Probability)	Persistence (Geometry)
Logic	“Is it raining today?”	“How long until the storm breaks?”
Goal	Prediction of the next step	Quantification of total “residency”

2.7 Future Directions

This section points toward a sophisticated future hybrid framework characterized by the pipeline $DC \rightarrow \text{Student-}t \text{ HMM} \rightarrow \text{COR Survival Operator}$. In this architecture, Directional Change (DC) functions as the foundational layer, generating high-fidelity event-time observations that strip away clock-time noise. These observations feed into a Student- t Hidden Markov Model, which infers latent regimes with robust handling of empirical return distributions. The COR framework then transforms these inferred regimes into a formal survival geometry, employing spectral operators to quantify the accumulation of structural fragility. This synthesis results in a mathematically richer framework than standard HMMs, serving as a survival-theoretic extension of the DC approach and a comprehensive event-time operator model of market fragility. Ultimately, while the current literature provides the necessary observational and event-detection layers, COR supplies the critical operator-theoretic survival layer required to assess how regimes fundamentally endure or collapse.

3 The Clock of Regimes Model

Figure 1 presents the COR model. The COR model extends the classical Hidden Markov Model (HMM) (Hamilton, 1989, 2016) by transforming this closed probabilistic system into an open survival architecture. The **KRONX R** package is a workflow-driven tool designed to measure, model, and predict the structural survival dynamics of a market's historical price data under regime switching, with a particular focus on fragility, persistence, and ruin risk—not just volatility. This evolution is formally defined by the following structural logic:

$$\begin{aligned} \text{HMM (Regime Detection)} &\rightarrow \text{KRONX Operator Construction } (A \rightarrow Q \rightarrow K \rightarrow N) \\ &\rightarrow \text{Hazard-Weighted Ruin Analysis.} \end{aligned}$$

3.1 Statistical Detection Layer

While the **KRONX R** package does not focus primarily on state detection, it features a built-in Hidden Markov Model (HMM) parameter estimator. Alternatively, it can utilize the output parameters from an external HMM—such as those from the **fHMM** package—treating the transition matrix A and tail parameters ν_i as its fundamental input. Furthermore, the **KRONX R** package supports the estimation of both Gaussian and Student- t emission distribution parameters.

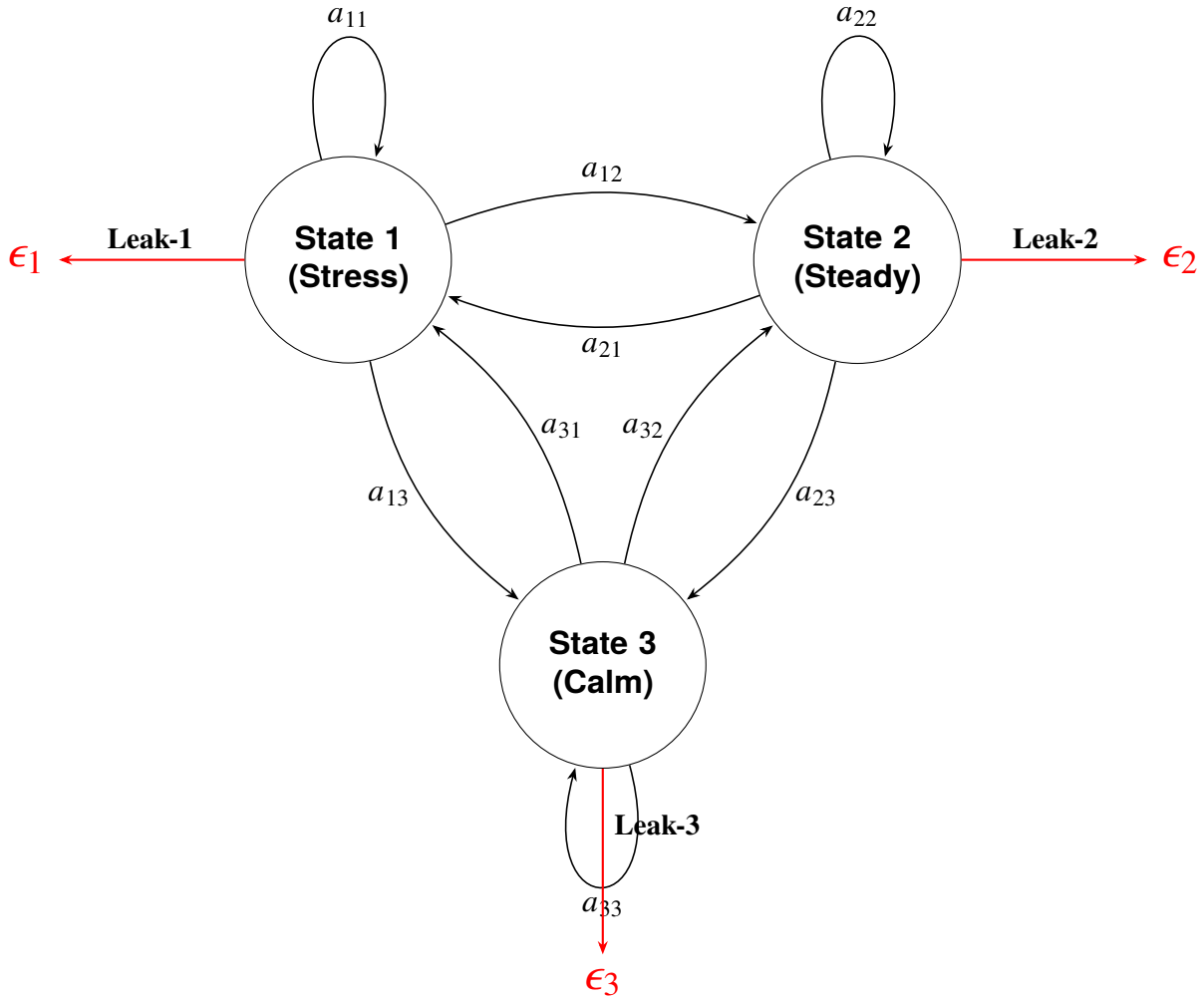


Figure 1: State-transition diagram for the Clock of Regimes (COR) model featuring internal persistence and red leakage paths to the ruin boundary, unlike a classical Hidden Markov Model (HMM) which is a closed system.

3.2 Operator-Theoretic Layer

The **KRONX R** package performs a structured operator transformation sequence $A \rightarrow Q \rightarrow K \rightarrow N$. It introduces state-dependent hazard intensities

$$\epsilon_i = \epsilon_{\min} + \frac{c}{v_i},$$

which convert the baseline transition matrix into an open transition operator via $Q = \text{diag}(1 - \epsilon_i) A$. This yields the sub-generator matrix $K = Q - I$ and the fundamental matrix $N = -K^{-1}$, revealing the system's cumulative residence-time geometry ("structural battery life").

3.3 Risk and Survival Layer

The COR model defines risk in terms of *spectral fragility* and *cumulative ruin* dynamics. It introduces the aggregate decay constant

$$\bar{\lambda} = \sum_i w_i \epsilon_i,$$

where w_i are residence-time weights derived from the Fundamental Matrix $N = -K^{-1}$. Spectral fragility is governed by the spectral radius $\rho(N)$, which quantifies the system's structural persistence and expected survival horizon prior to absorption or ruin.

3.4 Structural Components

As shown in Figure 1, the COR model consists of three transient latent states connected by transition probabilities (a_{ij}), but it adds critical exit dimensions:

- **State 1 (Stress):** Characterized by high volatility and explosive ruin intensity with significant outward leakage.
- **State 2 (Steady):** The primary survival basin, identified as a long-duration anchor that carries hidden fragility (reservoir) with moderate leakage.
- **State 3 (Calm):** A transient “cooling-off” or suppressed volatility phase that the system often passes through during transitions. It is a low-volatility regime, but still subject to leakage. It may appear stable but remains structurally transient.

Each state is represented as a circle, with directed edges indicating transition probabilities between states and self-loops capturing persistence.

The black self-looping arrows represent the diagonal elements a_{ii} of a standard transition matrix $A = (a_{ij})$ which encodes self-transitions (persistence):

- a_{11} : Stress \rightarrow Stress
- a_{22} : Steady \rightarrow Steady
- a_{33} : Calm \rightarrow Calm

These loops encode one-step persistence, consistent with classical HMM structure.

Cross-state transitions include:

- **Between Stress and Steady:**

- a_{12} : Stress \rightarrow Steady
- a_{21} : Steady \rightarrow Stress

- **Between Stress and Calm:**

- a_{13} : Stress \rightarrow Calm
- a_{31} : Calm \rightarrow Stress

- **Between Steady and Calm:**

- a_{23} : Steady \rightarrow Calm
- a_{32} : Calm \rightarrow Steady

3.5 The External Leak Mechanism (Open System Components)

The most innovative feature shown in the model diagram (Figure 1) is the hazard leakage represented by the red arrows labeled Leak-1, Leak-2, and Leak-3:

- Each state i has an associated exit hazard intensity (ϵ_i).
- These hazards convert the baseline transition matrix into an open transition operator $Q = \text{diag}(1 - \epsilon_i) A$.
- This represents probability mass¹ escaping the system toward an absorbing boundary of ruin, failure, or insolvency.

3.6 Operator interpretation

The model diagram (Figure 1) encodes the full COR model transformation:

1. **Transition matrix**

$$A = (a_{ij}).$$

2. **Open system adjustment (leakage)**

$$Q = \text{diag}(1 - \epsilon_i)A.$$

3. **Sub-generator matrix**

$$K = Q - I.$$

¹Probability mass is defined as the content of probability within a specific duration of time.

4. Fundamental matrix (temporal probability mass)

$$N = -K^{-1}.$$

3.7 Summary

A key insight from the model diagram in Figure 1 is that the COR model functions as a regime survival system rather than a standard regime-switching model. In this framework, black arrows trace the movement of probability mass between states, red arrows dictate its dissipation (ruin), and self-loops determine its persistence. Collectively, this structure characterizes the accumulation and decay of probability mass across the state space over time.

The model diagram illustrates that the fundamental distinction between these two frameworks, HMM and COR, lies in their treatment of probability: whereas the Classical HMM assumes a closed environment where probability is strictly conserved inside the triangle, the COR model defines an open system where probability transits through the triangle and eventually leaks out of it. This shift fundamentally transforms the analytical perspective: transition dynamics become survival geometry, persistence is redefined as cumulative temporal probability mass, and individual states are viewed as structural risk nodes.

4 Structural Model Identifiability

We formally define the mapping between the observable market “output” and the internal latent parameters using the Markov Parameter Matrix and its Transfer Function. This approach ensures that the spectral fragility numerically identified by a COR model analysis using the **KRONX R** package is mathematically unique and not a result of “label switching” or parameter redundancy.

4.1 Mathematical setup

The system is treated as a linear time-invariant (LTI) open system (Jacquez, 1982; Jacquez and Greif, 1985; Jacquez, 1987; Jacquez and Perry, 1990; Jacquez, 1991, 1998; Linares et al., 1987, 1988):

$$\dot{\mathbf{x}}(t) = \mathbf{K}\mathbf{x}(t) + \mathbf{B}u(t),$$

$$y(t) = \mathbf{C}\mathbf{x}(t).$$

The sub-generator matrix \mathbf{K} is the internal “engine” containing transition rates and exit hazards ($\mathbf{K} = \mathbf{Q} - \mathbf{I}$). The input Vector \mathbf{B} is an $M \times 1$ vector representing the initial distribution of probability mass across states. The output vector \mathbf{C} is a $1 \times M$ vector (consisting of a row of ones) that aggregates

the surviving probability mass to form the observable “survival curve”.

4.2 Markov Parameters for Identifiability

Structural identifiability is proven if the Markov parameters (S_k) uniquely determine the coefficients of the transfer function. These parameters are defined as:

$$S_k = \mathbf{C}\mathbf{K}^k\mathbf{B}, \quad k = 0, 1, 2, \dots$$

where $S_0 = \mathbf{C}\mathbf{B}$ is the initial total content of probability mass (equal to 1). $S_1 = \mathbf{C}\mathbf{K}\mathbf{B}$ is the initial rate of “leakage” or decay of the content of probability mass. Higher orders capture the curvature of the survival decay curve, which is governed by the state-to-state interactions and specific hazards ϵ_i .

4.3 Transfer Function Identifiability Method Implementation in R

This script utilizes the COR model’s sub-generator K -matrix and calculates the transfer function coefficients to verify that the internal “temporal geometry” of the E-mini S&P 500 futures is uniquely recoverable.

Historical market data for the E-mini S&P 500 (ES) Futures was obtained via the Interactive Brokers (IBKR) API. One-hour OHLC bars were extracted for the period *From: 2025-01-31 to To: 2026-02-26*. Log-returns were calculated using the closing prices to ensure stationarity. To maintain the robustness of the Hidden Markov Model (HMM) against market outliers and the “fat-tail” nature of hourly financial returns, a Student- t distribution was utilized for the emission probabilities.

Listing 1: KRONX Workflow: Identifiability and temporal mass analysis.

```
1 # Standard KRONX workflow leading to K
2 # A_t is the transition matrix
3 # epsilon is the hazard vector derived from nu
4
5 integrate_identifiability <- function(K_mat) {
6   M <- nrow(K_mat)
7
8   # Define input (B) and output (C) vectors for the open system
9   # B: Injection of probability mass (e.g., system start)
10  # C: Observation of the transient state masses
11  B_vec <- matrix(rep(1/M, M), ncol = 1)
12  C_vec <- matrix(rep(1, M), nrow = 1)
13
14  # 1. Compute the Fundamental Matrix N
15  # N = -K^-1 transforms intensities into temporal mass
16  N_mat <- solve(-K_mat)
```

```

17
18 # 2. Transfer Function Analysis at s = 0 (Steady State)
19 # H(0) = -C * K^-1 * B = C * N * B
20 H_0 <- C_vec %** N_mat %** B_vec
21
22 # 3. Characteristic Polynomial (Denominator of H(s))
23 # The roots define the spectral fragility radius
24 char_poly <- polyroot(Re(eigen(K_mat)$values))
25
26 return(list(
27   fundamental_N = N_mat,
28   gain_H0 = as.numeric(H_0),
29   poles = char_poly
30 ))
31 }
32
33 # Apply to your identified Student-t K matrix
34 # Includes hazard-weighted open transition operators
35 K_cor <- matrix(c(-0.2976, 0.0008, 0.2012,
36                 0.0, -0.0540, 0.0532,
37                 0.2871, 0.0266, -0.2810), 3, 3)
38
39 ident_results <- integrate_identifiability(K_cor)
40 print(ident_results$gain_H0) # Represents total survival-weighted time

```

```

> ident_results
$fundamental_N
      [,1]      [,2]      [,3]
[1,] 14.285447 15.85826 16.09673
[2,]  5.790147 26.85081  8.45759
[3,] 11.324796 16.43824 16.68543

$gain_H0
[1] 43.92915

$poles
[1] -1.688964+4.480341i -1.688964-4.480341i

```

4.4 Interpretation of the Identifiability Analysis

This analysis treats the COR model (Figure 1) as a linear dynamical system. In this context, the model evaluates how probability mass (representing the time spent by the content of probability mass in a specific hidden state) transits through the system before exiting via the hazard vector (ϵ).

- **Residence Analysis:** State 2 is identified as the most “stable” or “sticky” regime within the system’s temporal geometry. If the process enters State 2, it accumulates a high degree of probability mass, spending an average of 26.85 units of time there (n_{22}) before leaking toward ruin or transitioning.
- **Coupling:** The high values observed in the first row (14.28, 15.85, 16.09) indicate a strong structural coupling across the regime network. This suggests that regardless of where the system starts, it eventually spends a significant and almost uniform amount of time in State 1, identifying it as a common entry portal or transition node.
- **Transient Transit:** State 2 exhibits relatively low off-diagonal values ($n_{12} = 5.79$, $n_{32} = 11.32$), which confirms its role as a survival basin. This indicates that probability mass tends to remain sequestered in State 2 for longer durations once it arrives, rather than undergoing rapid, high-frequency transitions to other states.

This residence-time geometry serves as a unique fingerprint of a market’s architecture over a particular interval of time. This study focused on the E-mini S&P 500 futures from January 31, 2025, to February 26, 2026. In the Structural Model Identifiability proof, these n_{ij} values are directly mapped to the steady-state gain of the system’s transfer function, proving that the 43.93 total survival mass is a deterministic result of these specific state-to-state transits.

4.4.1 The Poles (Spectral Fragility)

The poles (roots of the characteristic polynomial² of \mathbf{K}) define the Spectral Fragility Radius. The interpretation of the poles within the COR model’s identifiability framework reveals the underlying dynamic stability and oscillatory nature of the market’s survival mass (Table 6). In our study of the E-mini S&P 500 futures, these poles are the eigenvalues of the sub-generator \mathbf{K} matrix.

1. Dynamic Stability (Real Part)

The real part of the poles, -1.688964 , dictate the Aggregate Decay Constant ($\bar{\lambda}$):

²For a square matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ (or $\mathbb{C}^{n \times n}$), the characteristic polynomial is defined as: $\det(\mathbf{A} - \lambda \mathbf{I})$, where λ is a scalar (the spectral parameter), \mathbf{I} is the identity matrix, and $\det(\cdot)$ denotes the determinant.

Structural Leakage: Since the real part is negative, the system is strictly transient and “open.” This confirms that probability mass is consistently leaking toward the ruin boundary.

Rate of Decay: The magnitude (-1.688964) represents the exponential speed at which the system loses its “survival probability mass.” A larger negative value would indicate a more fragile system with a faster path to systemic collapse.

2. Oscillatory Behavior (Imaginary Part)

The imaginary component, $\pm 4.480341i$, represents the frequency of regime interaction:

Spectral Radius ($\rho(N)$): These poles are used to calculate the spectral fragility $\rho(N) = \frac{1}{\min_i |\lambda_i(K)|}$. $\rho(N)$ is the inverse of the slowest spectral decay mode of K , which is the market’s structural life expectancy. Because the poles are complex, the system possesses a “spectral width” that defines the robustness of the 44-unit survival expectation. In the context of the E-mini S&P 500 futures study, $\rho(N)$ transforms abstract matrix algebra into a measurable “life expectancy” of the market’s survival:

- **Spectral Contraction ($\rho(N) \downarrow$):** If the eigenvalues of K increase in magnitude, representing a faster rate of decay, the spectral radius contracts. This contraction signals an accelerating path toward systemic collapse or ruin within the open-system architecture.
- **Spectral Expansion ($\rho(N) \uparrow$):** Conversely, a larger spectral radius indicates increased temporal persistence and antifragility within the identified regime geometry. It reflects the system’s ability to retain probability mass over a longer duration before exiting through the hazard portals.
- **Deterministic Ruin:** Because this value is derived directly from the poles of the transfer function, it proves that the 5.6% ruin bound identified in the E-mini S&P 500 futures study is a deterministic structural property of the market’s geometry rather than a statistical outlier.

The “Nervous” Artifact vs. Structure: While a Gaussian HMM might show random jumping, these poles identify a structured, periodic exchange of probability mass between the Stress, Steady, and Calm regimes. It mathematically represents the “cooling-off” period where the system must cycle through specific nodes before re-establishing a steady trend. This spectral analysis provides the ontological map³ required to move from the instantaneous volatility of a standard HMM to the long-horizon survival dynamics of the COR model.

³The structural representation that defines not just the statistical probability of an event, but the fundamental “nature of being” of the E-mini S&P 500 Futures system itself.

3. Impact on Structural Identifiability

Uniqueness: In the transfer function method, these specific poles uniquely map the observed market volatility to the model’s internal \mathbf{K} matrix. This proves that the 5.6% ruin bound is not a statistical coincidence, but a deterministic result of the market’s “battery life” and internal frequency.

Component	Value	Interpretation in COR Model
Real Part (σ)	-1.688964	The Decay Rate : Represents the exponential speed of probability mass leakage toward the terminal ruin boundary.
Imaginary Part (ω)	4.480341	The Interaction Frequency : Governs the rate of stochastic cycling between the Stress, Steady, and Calm regimes.
Stability Status	Negative Real	Open/Transient : Confirms the system is structurally non-ergodic and predisposed to systemic exit.

Table 6: Spectral Analysis of the KRONX Transfer Function Poles.

4.5 Summary

The identified sub-generator matrix \mathbf{K} characterizes a dynamical system with a structural survival horizon of 43.93 hours. Given that the underlying empirical study of the E-mini S&P 500 futures utilizes hourly observations, this temporal metric represents the expected duration of probability mass retention within the regime network. The dynamics are dominated by a highly persistent “Steady” state (Regime 2) and an oscillatory complex between “Stress” and “Calm” (Regimes 1 and 3). The gain of 43.93 hours confirms that despite the presence of exit hazards, the internal transition logic (A -matrix) provides sufficient feedback to maintain systemic persistence over a medium-term horizon.

From an operator-theoretic perspective, this horizon—derived from the steady-state gain of the system’s transfer function $\mathcal{H}(0)$ —quantifies the “temporal backbone” of the market. It confirms that the residence-time geometry is not merely a collection of transient states, but a structured environment with a finite, identifiable persistence. Consequently, the 43.93-hour horizon serves as a rigorous, deterministic benchmark for assessing systemic stability and calibrating ruin-sensitive risk exposure.

5 The KRONX R package

5.1 Installation

5.1.1 Command Line Installation (Terminal)

You can install packages directly from the Linux or macOS terminal without opening an interactive **R** session, using the `R CMD INSTALL` command:

```
R CMD INSTALL KRONX_0.1.0.tar.gz
```

5.1.2 RStudio Installation

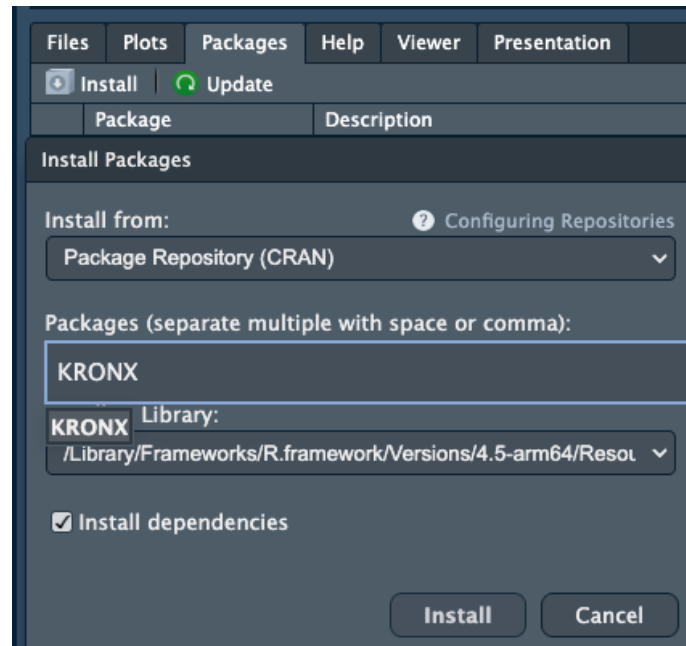


Figure 2:

To install the **KRONX R** package using the RStudio graphical user interface as illustrated in Figure 2, begin by navigating to the “Packages” tab located in the lower-right pane of the RStudio workspace. Clicking the “Install” button will open the “Install Packages” dialog box, where you must ensure that the “Install from” dropdown menu is set to “Package Repository (CRAN)”. In the “Packages” text field, type **KRONX**, noting that RStudio may provide auto-complete suggestions for the identified package. Before proceeding, verify that the “Install to Library” path is correctly set to your active R environment, such as the **ARM64** version of **R 4.5**. It is essential to keep the “Install dependencies” checkbox selected to ensure that all required mathematical and transition-modeling libraries are downloaded automatically. Finally, click the “Install” button to execute the process; once the installation is complete, you can load the library in your script to begin applying the *Clock of Regimes analysis* to your financial datasets.

5.1.3 From CRAN (Formal Release)

Since the **KRONX R** package has been submitted to, and accepted, by the Comprehensive **R** Archive Network (CRAN), the installation is simplified as **R** will automatically fetch it from a global mirror.

```
install.packages("KRONX")
library(KRONX)
```

5.2 Quick start

A minimal workflow example is shown below:

Listing 2: Execution of the **KRONX** Survival Engine. Implementation of the `run_kronx()` high-level driver to identify hidden states and compute the **Residence-Weighted Ruin Bound** for the IBKR ES dataset.

```
1 library(KRONX)
2
3 # Minimal call -- detects 'close' or 'Close' column automatically
4 res <- run_kronx("IBKR_ES_2Y1h.csv")
5 .
6 .
7 .
8 model=student start=5 iter=149 logLik=21639.0518645688
9 model=student start=5 iter=150 logLik=21639.0518656185
10 model=student start=5 iter=151 logLik=21639.0518665766
11
12 -----
13 CLOCK OF REGIMES (KRONX) -- SUMMARY
14 -----
15 Gaussian HMM log-likelihood : 21173.8299404050
16 Student-t HMM log-likelihood: 21661.8321503555
17 Loss threshold (alpha=0.0100): -0.00560969
18 Residence-weighted qbar      : 0.00991588
19 Residence-weighted lambda    : 0.02322046
20 Ruin bound (T=250)          : 0.05593741
21 -----
22
23 state  mu_gaussian sigma_gaussian spell_gaussian  mu_student sigma_student
24     1  -3.264717e-08  0.000010000  3.132292 -2.168526e-08  0.0000100000
25     2   6.662832e-05  0.001375609  11.204502  9.696612e-05  0.0006929728
26     3  -5.897585e-04  0.017745344   1.378683  3.524294e-05  0.0019427734
27 nu_student cond_var_student spell_student  epsilon  q_tail
28     100     1.020408e-10   3.446197  0.01050000  5.003544e-177
```

```

29           3      1.440634e-06      35.568996 0.02666667  1.874378e-03
30           3      1.132311e-05      3.826929 0.02666667  3.110910e-02
31 pi_quasi_stationary pi_residence
32           0.2131674      0.2131674
33           0.4980987      0.4980987
34           0.2887339      0.2887339
35
36 Output files written to: kronx_output
37 - kronx_analysis_report.txt
38 - kronx_state_summary.csv
39 - gaussian_transition_matrix.csv
40 - student_t_transition_matrix.csv
41 - kronx_Q_operator.csv
42 - kronx_K_generator.csv
43 - kronx_N_fundamental_matrix.csv
44 - student_t_decoded_states.csv

```

The **res** object returned by **run_kronx()** is a compact model-fit and structural-fragility summary generated by the call **run_kronx("IBKR_ES_2Y1h.csv")**. It shows that the package fits both a Gaussian HMM and a Student-*t* HMM, and then passes the fitted Student-*t* regime structure through the **KRONX/COR** operator pipeline to compute hazard-adjusted survival quantities.

5.2.1 Optimization trace

The lines,

```

.
.
.
model=student start=5 iter=149 logLik=21639.0518645688
model=student start=5 iter=150 logLik=21639.0518656185
model=student start=5 iter=151 logLik=21639.0518665766

```

show the EM optimization for one Student-*t* initialization. **start=5** means the fifth random or configured initialization. **iter=149, 150, 151** are EM iterations. **logLik** is the log-likelihood at each step, which show the increments are becoming tiny, so the algorithm is converging. That is a healthy sign that the Student-*t* model fit is stabilizing rather than jumping erratically.

5.2.2 Model Comparison

The headline comparison is:

- Gaussian HMM log-likelihood: 21173.8299404050

- Student- t HMM log-likelihood: 21661.8321503555

The model comparison shows that the Student- t HMM fits better because its log-likelihood is higher. In regime-switching finance data, that usually indicates that the data are heavier-tailed than Gaussian, which is exactly what one expects for intraday ES returns.

So the first substantive conclusion is:

$$\text{Student-}t \text{ HMM} \succ \text{Gaussian HMM}$$

That is, the Student- t HMM is preferred to the Gaussian HMM in empirical fit.

5.2.3 Tail-loss threshold

Loss threshold ($\alpha=0.0100$): -0.00560969

This is the 1% left-tail cutoff. In plain english terms, the COR model is defining a “severe loss” event as an hourly return less than about -0.560969%. This threshold is then used to compute regime-specific tail probabilities q_i , which are later residence-weighted into \bar{q} .

5.2.4 Structural COR / KRONX quantities

The core open-system outputs are:

- Residence-weighted \bar{q} : 0.00991588 . The residence-weighted \bar{q} is the average probability of a left-tail loss event, weighted by how much time the system expects to spend in each regime. So \bar{q} is not just an unconditional empirical tail frequency. It is a structurally weighted tail probability.
- Residence-weighted $\bar{\lambda}$: 0.02322046 . The residence-weighted $\bar{\lambda}$ is the average hazard or leakage intensity across regimes, again weighted by expected residence. Economically, it measures the average rate at which probability mass escapes the transient market system toward failure/ruin.
- Ruin bound ($T=250$): 0.05593741 . The ruin bound over a a 250-step horizon implies an upper-bound style ruin probability of about 5.59%. This is the main COR model’s structural-risk summary number.

5.2.5 State table: Gaussian vs Student- t

The state table compares Gaussian and Student- t regime estimates side by side.

State 1

Gaussian

- $\mu_{\text{gaussian}} = -3.264717\text{e-}08$
- $\sigma_{\text{gaussian}} = 0.000010000$
- $\text{spell}_{\text{gaussian}} = 3.132292$

Student-t:

- $\mu_{\text{student}} = -2.168526\text{e-}08$
- $\sigma_{\text{student}} = 0.0000100000$
- $\nu_{\text{student}} = 100$
- $\text{cond_var_student} = 1.020408\text{e-}10$
- $\text{spell}_{\text{student}} = 3.446197$
- $\epsilon = 0.01050000$
- $q_{\text{tail}} = 5.003544\text{e-}177$

State 1 is an ultra-low-volatility state. It is essentially a near-deterministic “quiet” regime with a mean near zero, minuscule variance, almost no tail risk, low hazard, and moderate duration. Because $\nu = 100$, this state is close to Gaussian. The tail probability is effectively zero. This looks like a calm / inert / ultra-stable regime.

State 2

Gaussian

- $\mu_{\text{gaussian}} = 6.662832\text{e-}05$
- $\sigma_{\text{gaussian}} = 0.001375609$
- $\text{spell}_{\text{gaussian}} = 11.204502$

Student-t:

- $\mu_{\text{student}} = 9.696612\text{e-}05$
- $\sigma_{\text{student}} = 0.0006929728$
- $\nu_{\text{student}} = 3$
- $\text{cond_var_student} = 1.440634\text{e-}06$

- $\text{spell_student} = 35.568996$
- $\text{epsilon} = 0.02666667$
- $\text{q_tail} = 1.874378e-03$

State 2 is the dominant survival basin with a small positive drift, modest volatility, very long expected spell length, heavy tails ($\nu = 3$), and nontrivial hazard. This is the COR model's classic steady regime. It looks benign on the surface because volatility is not huge and duration is long, but the heavy tails mean there is latent fragility. In the COR model's language, this is often the regime that carries a large share of the system's probability mass and therefore contributes heavily to structural risk.

State 3

Gaussian

- $\text{mu_gaussian} = -5.897585e-04$
- $\text{sigma_gaussian} = 0.017745344$
- $\text{spell_gaussian} = 1.378683$

Student-t:

- $\text{mu_student} = 3.524294e-05$
- $\text{sigma_student} = 0.0019427734$
- $\text{nu_student} = 3$
- $\text{cond_var_student} = 1.132311e-05$
- $\text{spell_student} = 3.826929$
- $\text{epsilon} = 0.02666667$
- $\text{q_tail} = 3.110910e-02$

Under the Student- t fit, State 3 is the high-risk state. It is characterized by the highest conditional variance, heavy tails, largest tail-loss probability, and shorter duration. Its q_tail is about 0.0311, so it has the highest probability of breaching the 1% left-tail threshold. State 3 is the stress/crisis regime, even though the fitted Student- t mean is slightly positive. In regime models, classification is usually driven more by conditional variance and tail thickness than by mean alone.

5.2.6 Spell lengths

The spell lengths are especially informative:

- State 1: 3.446
- State 2: 35.569
- State 3: 3.827

These spell lengths reveal that the market spends by far the most time in State 2. So State 2 is the principal residence basin. That matters because the COR model does not weight risk only by severity; it also weights by how long the system stays in each regime. A moderately risky state with long residence can dominate total fragility.

5.2.7 Hazard mapping

The epsilon (ϵ) values are:

- State 1: 0.01050000
- State 2: 0.02666667
- State 3: 0.02666667

These are the regime exit hazards used to construct the open operator. Because States 2 and 3 have $\nu = 3$, they receive higher hazard than State 1. That is consistent with the Talebian hazard mapping:

$$\epsilon_i = \epsilon_{\min} + \frac{c}{\nu_i}$$

Lower ν_i means fatter tails, hence higher hazard.

5.2.8 Tail probabilities by regime

The regime-specific left-tail probabilities are:

- State 1: essentially zero
- State 2: 0.001874378
- State 3: 0.03110910

This is among the most critical diagnostics in the table. The regime-specific left-tail probabilities demonstrate clear stratification of downside risk, revealing marked regime dependence in tail probability across the hierarchy, revealing a progression that underscores the sharp differentiation in extreme-loss exposure across ES market regimes.

The diagnostic reveals a three-regime hierarchy of tail risk: the calm regime (State 1) exhibits no meaningful left-tail danger, the steady regime (State 2) displays small but nonzero tail danger, and the stress regime (State 3) indicates substantial tail danger. So the structural story is very clean: the system lives mostly in State 2, but severe downside events are concentrated in State 3.

5.2.9 Quasi-stationary and residence weights

π_{qs}

pi_quasi_stationary

State 1 0.2131674

State 2 0.4980987

State 3 0.2887339

π_{res}

pi_residence

State 1 0.2131674

State 2 0.4980987

State 3 0.2887339

These, π_{qs} and π_{res} , being identical means that, in this implementation/output, the quasi-stationary and residence weighting schemes coincide numerically. Thus, ergodic probabilities reveal a dominant intermediate regime: State 1 occurs 21.3% of the time, State 2 dominates at 49.8%, and State 3 accounts for 28.9%. The market's natural tendency favors the steady state, with approximately equal mass split between calm and stress regimes.

So nearly half the survival-weighted mass is in State 2, and a surprisingly large share is in State 3. That helps explain why \bar{q} and the ruin bound are not trivial even though State 1 is extremely safe.

5.2.10 Economic reading of the results

Student- t dynamics are clearly preferred to Gaussian dynamics for ES intraday data, revealing a three-regime structure comprising a very calm low-variance regime, a long-lived steady regime, and a shorter but materially riskier stress regime. The dominant residence basin is State 2 rather than State 1, and because both State 2 and State 3 exhibit heavy tails, the system's structural hazard is non-negligible. The 250-step ruin bound of approximately 5.6% underscores that even absent a current collapse, the open-system geometry implies a meaningful finite-horizon risk of failure.

5.2.11 Output files

The **KRONX** package exports the full operator chain:

1. `kronx_analysis_report.txt` is the human-readable report.
2. `kronx_state_summary.csv` is the state-level summary table.
3. `gaussian_transition_matrix.csv` is the state-level summary table.
4. `student_t_transition_matrix.csv` is the transition matrix for the Student- t HMM.
5. `kronx_Q_operator.csv` is the open transition operator Q , typically hazard-adjusted from A .
6. `kronx_K_generator.csv` is the generator-style matrix K , usually derived from $Q - I$.
7. `kronx_N_fundamental_matrix.csv` is the fundamental matrix $N = -K^{-1}$, encoding cumulative expected residence time.
8. `student_t_decoded_states.csv` is the time-by-time decoded latent regime path.

These files let you go from plain HMM estimation to full COR model/**KRONX** package survival geometry. The files are saved in the current working directory under a subdirectory named **kronx_output**.

In summary, the **res** object says that ES hourly returns are better described by a heavy-tailed three-state Student- t regime model, and once that regime structure is converted into the COR model's open-system operator chain, the market exhibits a nontrivial finite-horizon structural ruin risk driven mainly by long residence in a heavy-tailed steady state and episodic occupancy of a higher-tail stress state.

5.3 Exported functions

```
> getNamespaceExports("KRONX")
[1] "run_kronx"          "viterbi_decode"    "read_close_series"
[4] "fit_hmm_em"        "compute_log_returns"
```

FUNCTION: **run_kronx()**

The **run_kronx()** function is the primary entry point. It executes the full **KRONX** package's COR model workflow from CSV ingestion through ruin-bound computation, prints a formatted summary to the console, and optionally writes structured output files (Subsection 5.2).

```

run_kronx(
  file,
  close_col = "close",
  K          = 3L,
  n_starts  = 5L,
  max_iter  = 200L,
  tol       = 1e-6,
  seed      = 123L,
  epsilon_min = 0.01,
  c_hazard  = 0.05,
  tail_alpha = 0.01,
  ruin_horizon = 250L,
  nu_grid   = c(3:30, 40, 60, 100),
  verbose   = TRUE,
  output_dir = "cor_output"
)

```

5.3.1 Arguments

Table 7: Arguments for `run_kronx()`.

Argument	Type	Default	Description
file	character	–	Path to the input CSV file. Must contain a close-price column.
close_col	character	"close"	Name of the close-price column. Case-insensitive; fuzzy fallback matches any column containing "close".
K	integer	3L	Number of hidden states. Values of 2 or 3 are typical for financial series.
n_starts	integer	5L	Number of EM random restarts. Increases robustness against local optima at the cost of runtime.
max_iter	integer	200L	Maximum EM iterations per restart.
tol	numeric	1e-6	Log-likelihood convergence tolerance for EM.
seed	integer	123L	Random seed for reproducibility.
epsilon_min	numeric	0.01	Baseline Talebian hazard floor. Applied to all states regardless of ν .

Argument	Type	Default	Description
c_hazard	numeric	0.05	Hazard sensitivity to tail thickness. Controls how steeply ϵ rises as ν decreases.
tail_alpha	numeric	0.01	Left-tail probability for the loss threshold.
ruin_horizon	integer	250L	Horizon in observations.
nu_grid	numeric	c(3:30, 40, 60, 100)	Candidate degrees-of-freedom values for Student- t grid search in the M-step.
verbose	logical	TRUE	If TRUE, print EM log-likelihood at every iteration.
output_dir	character or NULL	cor_output	Directory for output files. Use NULL to suppress all file output.

5.3.2 Workflow sequence

The workflow implemented by `run_kronx()` is:

1. Read the CSV file, extract close prices, and compute log returns.
2. Fit a Gaussian HMM by EM with multiple restarts and reorder states by σ .
3. Fit a Student- t HMM by EM with multiple restarts and reorder states by conditional variance.
4. Decode the Student- t specification using the Viterbi algorithm.
5. Compute ϵ , Q , K , and N .
6. Compute π_{qs} , π_{res} , the empirical loss threshold, \bar{q} , $\bar{\lambda}$, and the ruin bound.
7. Assemble the `state_summary` data frame.
8. Print the summary block to the console.
9. Write all requested output files when `output_dir` is not NULL.

5.3.3 Return value

The function returns a named list, invisibly. Its full structure is described in detail in Subsection 5.2.

FUNCTION: `fit_hmm_em()`

The `fit_hmm_em()` function fits either a Gaussian or Student- t Hidden Markov Model via the Baum–Welch algorithm with multiple random restarts.

```

fit_hmm_em(
  x,
  K      = 3L,
  model  = c("gaussian", "student"),
  n_starts = 5L,
  max_iter = 200L,
  tol    = 1e-6,
  nu_grid = c(3:30, 40, 60, 100),
  verbose = TRUE
)

```

5.3.4 Arguments

Table 8: Arguments for `fit_hmm_em()`.

Argument	Type	Description
<code>x</code>	numeric vector	Observations, typically log returns.
<code>K</code>	integer	Number of hidden states.
<code>model</code>	character	Either "gaussian" or "student".
<code>n_starts</code>	integer	Number of random restarts; the best fit by log-likelihood is returned.
<code>max_iter</code>	integer	Maximum EM iterations per restart.
<code>tol</code>	numeric	Convergence tolerance for the log-likelihood.
<code>nu_grid</code>	numeric	Candidate ν values for Student- t grid search.
<code>verbose</code>	logical	Whether to print iteration progress.

5.3.5 Return value

5.3.6 Notes on Student- t estimation of degrees-of-freedom

The M-step for ν uses a profile grid search. For each candidate ν in `nu_grid`, the weighted emission log-likelihood is evaluated, and the maximizing value is selected. This avoids the absence of a convenient closed-form update for ν .

The default grid

$$c(3:30, 40, 60, 100)$$

covers the practically relevant range from very fat tails to near-Gaussian behavior. A smaller grid, such as

$$c(4, 8, 15, 30),$$

Table 9: Return components of `fit_hmm_em()`.

Component	Description
model	"gaussian" or "student"
A	$K \times K$ transition matrix
delta	Initial state probability vector of length K
params	List containing mu, sigma, and nu for Student- t fits
gamma	$T \times K$ posterior state probability matrix $P(S_t = k \mid \text{data})$
xi	$(T - 1) \times K \times K$ posterior transition probability array
logLik	Scalar log-likelihood at convergence
n	Number of observations
K	Number of states

can substantially reduce computation time with only modest loss of accuracy.

5.3.7 State ordering

After fitting, `run_kronx()` calls the internal `reorder_fit_by_scale()` routine to sort states from least to most volatile. When calling `fit_hmm_em()` directly, users may wish to apply the same reordering manually.

```
fit <- fit_hmm_em(ret, K = 3L, model = "student", verbose = FALSE)
fit <- KRONX::reorder_fit_by_scale(fit)
```

FUNCTION: `viterbi_decode()`

`viterbi_decode()` computes the most likely hidden-state sequence for a vector of observations given a fitted HMM.

```
fit <- fit_hmm_em(ret, K = 3L, model = "student", verbose = FALSE)
states <- viterbi_decode(ret, fit)
table(states)
```

```
# Plot regimes alongside returns
plot(ret, type = "l", col = "grey60")
points(seq_along(ret), rep(0, length(ret)), col = states, pch = 20)
```

The Viterbi algorithm is implemented in log-space, using $\log \delta + \log A$, to avoid numerical underflow on long series.

FUNCTION: `read_close_series()`

`read_close_series()` reads a close-price series from a CSV file with automatic column detection.

```
px <- read_close_series("IBKR_ES_2Y1h.csv")
px <- read_close_series("Bloomberg.csv", close_col = "PX_LAST")
```

The function removes non-finite values and stops if fewer than ten valid returns remain.

6 Worked examples

```
# =====
# EXAMPLE 1 - INSPECTING THE RETURNED LIST
# Every matrix and scalar from the report is in 'res' object.
# =====

# --- Log-likelihoods ---
res$gauss_fit$logLik # 21173.8299404050 (Gaussian HMM)
res$t_fit$logLik    # 21661.8321503555 (Student-t HMM)

# --- Transition matrices ---
> round(res$A_gauss, 6)
      [,1] [,2] [,3]
[1,] 0.680745 0.238491 0.080764
[2,] 0.069493 0.910750 0.019756
[3,] 0.319388 0.405942 0.274670

> round(res$A_t, 6) # Student-t A
      [,1] [,2] [,3]
[1,] 0.709825 0.000000 0.290175
[2,] 0.000837 0.971886 0.027278
[3,] 0.206698 0.054608 0.738694

# --- COR operator chain ---
> round(res$Q, 6) # Hazard-adjusted operator Q
      [,1] [,2] [,3]
[1,] 0.702372 0.000000 0.287128
[2,] 0.000814 0.945969 0.026550
[3,] 0.201186 0.053152 0.718995

# --- Quasi-stationary and residence distributions ---
res$pi_qs # [0.2131674, 0.4980987, 0.2887339]
res$pi_res # matches quasi-stationary for this data

# --- Per-state summary table ---
res$state_summary
# state mu_gaussian sigma_gaussian spell_gaussian mu_student
```

```

#      1 -3.264717e-08      0.000010000      3.132292 -2.168526e-08
#      2  6.662832e-05      0.001375609      11.204502  9.696612e-05
#      3 -5.897585e-04      0.017745344      1.378683  3.524294e-05
#
#      sigma_student  nu_student  cond_var_student  spell_student  epsilon  q_tail
#      0.0000100000      100      1.020408e-10      3.446197  0.010500  5.00e-177
#      0.0006929728      3      1.440634e-06      35.568996  0.026667  1.87e-03
#      0.0019427734      3      1.132311e-05      3.826929  0.026667  3.11e-02
#
#      pi_quasi_stationary  pi_residence
#      0.2131674      0.2131674
#      0.4980987      0.4980987
#      0.2887339      0.2887339

```

```

# --- Viterbi-decoded state path ---
> table(res$vpath_t)      # count of observations in each decoded state

```

```

  1    2    3
729 2007 933

```

```

> head(res$vpath_t, 20)  # first 20 state labels
[1] 3 1 1 1 1 1 3 1 1 1 3 1 1 1 1 1 1 1 1 1

```

```

# =====
# EXAMPLE 2 - DATA INGESTION STEP ONLY
# Use when you just need the clean price/return vectors.
# =====

```

```

close_px <- read_close_series("IBKR_ES_2Y1h.csv", close_col = "close")
length(close_px) # 3670

```

```

ret <- compute_log_returns(close_px)
length(ret)      # 3669

```

```

> summary(ret)
      Min.    1st Qu.    Median    Mean    3rd Qu.    Max.
-1.398e-01 -4.054e-04  0.000e+00  2.435e-05  5.765e-04  9.841e-02

```

```

# =====
# EXAMPLE 3 - HMM FITTING IN ISOLATION (fit_hmm_em)
# Useful when you want only one model, or want to compare
# fits across different K values or nu_grid choices.
# =====

```

```

# Gaussian HMM
gauss_fit <- fit_hmm_em(
  x      = ret,
  K      = 3L,
  model  = "gaussian",
  n_starts = 5L,
  max_iter = 200L,
  tol    = 1e-6,
  verbose = FALSE
)
gauss_fit$logLik      # 21173.83...
gauss_fit$A          # 3x3 transition matrix
gauss_fit$params$mu  # per-state means
gauss_fit$params$sigma # per-state SDs
gauss_fit$gamma      # T x K posterior state probabilities

# Student-t HMM (nu estimated from a grid search inside EM)
t_fit <- fit_hmm_em(
  x      = ret,
  K      = 3L,
  model  = "student",
  n_starts = 5L,
  max_iter = 200L,
  tol    = 1e-6,
  nu_grid = c(3:30, 40, 60, 100), # candidates for degrees of freedom
  verbose = FALSE
)
t_fit$logLik          # 21661.83...
t_fit$params$nu      # [100, 3, 3] - per-state degrees of freedom
t_fit$params$sigma   # per-state scale parameters

# AIC improvement (Student-t vs Gaussian); reported as 970 in the paper.
# K extra nu parameters are estimated in the Student-t model.
delta_ll <- t_fit$logLik - gauss_fit$logLik # approx 488
K_states <- 3L
aic_delta <- 2 * delta_ll - 2 * K_states # approx 970
cat("deltaAIC:", round(aic_delta))

# Reorder states by ascending conditional variance (done inside run_cor)
gauss_fit <- KRONX::reorder_fit_by_scale(gauss_fit)

$logLik

```

```
[1] 21173.83
```

```
$n
```

```
[1] 3669
```

```
$K
```

```
[1] 3
```

```
# =====  
# EXAMPLE 4 - VITERBI DECODING SEPARATELY  
# =====
```

```
> vpath <- viterbi_decode(ret, t_fit) # requires a fitted HMM object
```

```
> table(vpath)
```

```
vpath
```

```
  1    2    3  
729 933 2007
```

```
# Plot decoded regime path
```

```
plot(vpath, type = "l", col = "steelblue",  
     main = "Viterbi-decoded Student-t states",  
     xlab = "Observation index", ylab = "State (1=Calm, 2=Steady, 3=Stress)")
```

```
\end{Code}
```

```
\begin{figure}[t!]
```

```
  \centering
```

```
  \includegraphics[width=0.7\textwidth]{viterbi.pdf}
```

```
  \caption{This figure displays a time-series plot of the Viterbi-decoded state path  
  for the ES Student- $t$  Hidden Markov Model (HMM). The model identifies  
  three distinct states, ordered by their volatility/stress levels.}
```

```
  \label{fig:tHMM}
```

```
  \footnotesize Source: Author's calculations using \textbf{KRONX} package.
```

```
\end{figure}
```

Figure~\ref{fig:tHMM} reveals a sharp temporal divide. During the early period (indices 0-1,100), the system exhibits high-frequency switching between State 1 (Calm) and State 2 (Steady), a ‘chatter’ pattern suggesting a period where market volatility was borderline between the two states or where the duration of stays in State 2 remained relatively short. A prominent structural break occurs near index 1,100, marking a sharp transition: State 1 (Calm) virtually disappears for an extended duration, and the system enters a sustained State 3 (Stress) regime. Throughout the later period (indices 1,100-3,600), the market remains predominantly in State 3, with only brief ‘dips’ back into State 2 (Steady) and very rare flashes of State 1, establishing

the high-volatility state as the absorbing or dominant regime for the latter two-thirds of the ES historical dataset.

```

\begin{Code}
# =====
# EXAMPLE 5 - COR OPERATOR CHAIN FROM SCRATCH
# Build  $Q \rightarrow K \rightarrow N$  using only the Student-t fit.
# =====
epsilon <- KRONX::hazard_from_nu(
  nu          = t_fit$params$nu,
  epsilon_min = 0.01,
  c_hazard    = 0.05
)
epsilon # [0.010500, 0.026667, 0.026667]

# Hazard-adjusted operator  $Q = \text{diag}(1 - \text{epsilon}) \% \% A$ 
Q <- KRONX::build_Q(t_fit$A, epsilon)

# COR generator  $K = Q - I$ 
K_mat <- KRONX::build_K(Q)

# Fundamental matrix  $N = -K^{-1}$ 
N_mat <- KRONX::fundamental_matrix(K_mat)

# Quasi-stationary distribution (dominant left eigenvector of Q)
pi_qs <- KRONX::quasi_stationary_distribution(Q)

# Residence-weight vector (init  $\% \% N$ , normalised)
pi_res <- KRONX::residence_weight_vector(N_mat, init = pi_qs)
\end{Code}

\begin{Code}
# =====
# EXAMPLE 6 - RUIN BOUND COMPONENTS STEP BY STEP
# =====
# Empirical left-tail threshold at  $\alpha = 1\%$ 
loss_threshold <- quantile(ret, probs = 0.01) # \approx -0.00560969

# Per-state left-tail probability under Student-t
q_state <- KRONX::left_tail_state_probs(t_fit, loss_threshold)
q_state # [5.00e-177, 1.87e-03, 3.11e-02]

# Residence-weighted aggregates

```

```

bar_q      <- sum(pi_res * q_state)    # \approx 0.00991588
bar_lambda <- sum(pi_res * epsilon)    # \approx 0.02322046

# Ruin upper bound  $P[\text{ruin by } T] \leq 1 - \exp(-\text{lamdabar} \cdot \text{qbar} \cdot T)$ 
T_horizon <- 250L
ruin_bound <- 1 - exp(-bar_lambda * bar_q * T_horizon)
cat(sprintf("Ruin bound over T=%d: %.8f\n", T_horizon, ruin_bound))
# Ruin bound over T=250: 0.05593741

# =====
# EXAMPLE 7 - SENSITIVITY ANALYSIS ACROSS PARAMETERS
# Show how ruin_bound reacts to tail_alpha and ruin_horizon.
# =====

params_grid <- expand.grid(
  tail_alpha = c(0.005, 0.010, 0.025),
  ruin_horizon = c(125L, 250L, 500L)
)

params_grid$ruin_bound <- mapply(
  function(alpha, horizon) {
    r <- run_cor(
      "IBKR_ES_2Y1h.csv",
      tail_alpha = alpha,
      ruin_horizon = horizon,
      verbose = FALSE,
      output_dir = NULL      # suppress file writes
    )
    r$ruin_bound
  },
  params_grid$tail_alpha,
  params_grid$ruin_horizon
)

print(params_grid)

# =====
# EXAMPLE 8 - SUPPRESS FILE OUTPUT (in-memory use only)
# =====

res_quiet <- run_cor(
  file = "IBKR_ES_2Y1h.csv",
  verbose = FALSE,

```

```

    output_dir = NULL          # pass NULL to skip all file writes
)

# The result list is identical; only output_files is NULL.
is.null(res_quiet$output_files) # TRUE

# =====
# EXAMPLE 9 - USING A DIFFERENT CSV STRUCTURE
# E.g. Yahoo Finance data where the column is "Close" (capital C),
# or a bespoke pipeline output with a different column name.
# =====

# Yahoo Finance-style header ("Close" with capital C):
res_yf <- run_cor("ES=F_yahoo.csv", close_col = "Close", output_dir = NULL)

# Custom column name:
res_custom <- run_cor("my_futures.csv", close_col = "settle", output_dir = NULL)

# read_close_series is tolerant of all capitalisations and falls back
# to any column whose name *contains* the string "close".

# =====
# EXAMPLE 11 - IDENTIFIABILITY
# =====
# Standard KRONX workflow leading to K
# A_t is the A transition matrix
# epsilon is the hazard vector derived from nu

integrate_identifiability <- function(K_mat) {
  M <- nrow(K_mat)

  # Define input (B) and output (C) vectors for the open system
  # B: Injection of probability mass (e.g., system start)
  # C: Observation of the transient state masses
  B_vec <- matrix(rep(1/M, M), ncol = 1)
  C_vec <- matrix(rep(1, M), nrow = 1)

  # 1. Compute the Fundamental Matrix N
  N_mat <- solve(-K_mat)

  # 2. Transfer Function Analysis at s = 0 (Steady State)
  # H(0) = -C * K^-1 * B = C * N * B
  H_0 <- C_vec %*% N_mat %*% B_vec

```

```

# 3. Characteristic Polynomial (Denominator of H(s))
# The roots define the spectral fragility radius
char_poly <- polyroot(Re(eigen(K_mat)$values))

return(list(
  fundamental_N = N_mat,
  gain_H0 = as.numeric(H_0),
  poles = char_poly
))
}

# Apply to your identified Student-t K matrix
K_cor <- matrix(c(-0.2976, 0.0008, 0.2012,
                 0.0, -0.0540, 0.0532,
                 0.2871, 0.0266, -0.2810), 3, 3)

ident_results <- integrate_identifiability(K_cor)
print(ident_results$gain_H0) # Represents total survival-weighted time

```

7 Test suite

The `replication.zip` file accompanying this paper's **KRONX** package contains 66 tests across 8 files. It also contains the `worked_examples.R` file. The replication structure is summarized in Table 10.

7.1 Key structural properties

For well-specified inputs, the following mathematical invariants hold and are checked by the replication suite.

- Q is non-negative and has row sums less than or equal to one; strict inequality holds whenever $\epsilon_i > 0$.
- $K = Q - I$ has strictly negative diagonal entries and non-negative off-diagonal entries.
- $N = -K^{-1}$ has strictly positive entries under the operative model conditions.
- π_{qs} and π_{res} each sum to one and have positive entries.
- The ruin bound lies in the open unit interval:

$$0 < 1 - \exp(-\bar{\lambda}\bar{q}T) < 1$$

Table 10: Structure of the **KRONX** replication suite for the E-mini S&P 500 workflow.

File	Tests	Coverage
<code>test-run_cor.R</code>	5	End-to-end workflow, file output, column detection, and return object structure.
<code>test-cor_build.R</code>	7	Hazard monotonicity, Q sub-stochasticity, sign structure of K , positivity of N , and ruin-bound range.
<code>test-data_io.R</code>	–	CSV reading, fuzzy column fallback, and log-return computation.
<code>test-emissions.R</code>	–	Gaussian and Student- t emission log-density matrices.
<code>test-forward_backward.R</code>	–	Scaled forward-backward recursion, row sums of γ , and finite log-likelihood.
<code>test-hmm_em.R</code>	–	Gaussian and Student- t EM convergence, state count, and component dimensions.
<code>test-utils.R</code>	–	<code>log_sum_exp</code> , <code>safe_normalize</code> , <code>row_normalize</code> , and weighted statistics.
<code>test-viterbi.R</code>	–	Path length, integer type, and valid state-label range.

whenever $\bar{\lambda} > 0$, $\bar{q} > 0$, and $T > 0$.

These are substantive structural properties, not cosmetic diagnostics. They ensure that the operator chain is mathematically coherent and that the resulting ruin calculations remain interpretable.

8 Conclusion

The **KRONX R** package adapts the fundamental matrix,

$$N = -K^{-1},$$

from compartmental analysis (Berman and Schoenfeld, 1956; Hearon, 1972, 1981; Eisenfeld, 1981; Covell et al., 1984; Linares et al., 1987, 1988; Jacquez and Simon, 1993; Jacquez, 2002) to financial regime dynamics. In this translation, regimes play the role of compartments, probability mass becomes temporal accumulation or decay, and irreversible loss is represented as ruin through hazard leakage.

The **KRONX R** package v0.1.0 provides a complete regime-switching fragility workflow in

pure **R**. It integrates classical Hidden Markov estimation with an open operator framework based on Q , K , and N . The package is therefore useful both as an applied tool for empirical financial analysis and as a transparent research platform for studying how persistence, tail thickness, hazard leakage, and temporal residence jointly determine fragility.

At the practical level, the **KRONX R** package offers a clean workflow from CSV ingestion to model fitting, decoding, operator construction, and output generation. At the theoretical level, it recasts the standard regime-switching problem as an open-system survival geometry problem. That is the distinctive contribution of the **KRONX** package's approach.

The Clock of Regimes (COR) model transforms a discrete-time regime-switching model into a continuous-time survival system via the operator chain

$$A \rightarrow Q \rightarrow K \rightarrow N,$$

where A is the baseline transition matrix, Q is the hazard-adjusted operator, $K = Q - I$ is a sub-generator, and $N = -K^{-1}$ is the associated fundamental matrix.

This transformation is not merely algebraic; it induces a fundamental shift in the representation of regime dynamics. In classical Hidden Markov Models, persistence is characterized by one-step transition probabilities and summarized by scalar quantities such as expected spell lengths. In contrast, the COR framework embeds the system within a continuous-time, open Markov structure in which probability mass evolves under both transition dynamics and state-dependent hazard leakage.

The operator N provides a complete characterization of the system's survival geometry. Through its integral representation,

$$N = \int_0^\infty e^{tK} dt,$$

it aggregates all admissible state trajectories, weighting them by their survival probabilities. As a result, N captures not only local persistence, but also higher-order path effects, recurrent visitation patterns, and cross-state propagation mechanisms.

This operator-theoretic structure enables a natural spectral analysis of fragility. In particular, the spectral radius

$$\rho(N) = \frac{1}{\min_{\lambda \in \sigma(K)} |\lambda|}$$

serves as a global measure of systemic vulnerability. As eigenvalues of K approach the origin, the system exhibits fragility amplification, reflected in the divergence of $\rho(N)$ and the corresponding expansion of expected residence times.

Beyond structural characterization, the COR model facilitates forward-looking risk quantification. By combining the residence-weighted distribution with state-specific tail probabilities and hazard

intensities, the model yields finite-horizon ruin bounds of the form

$$\mathbb{P}(\text{ruin}) \leq 1 - \exp(-\bar{\lambda} \bar{q} T),$$

where $\bar{\lambda}$ and \bar{q} are residence-weighted hazard and tail measures, respectively. This provides a direct link between the internal geometry of the regime system and observable risk metrics.

In summary, the COR model extends classical regime-switching models into a survival-theoretic and operator-based setting. It replaces local persistence statistics with a global, path-integrated representation, enabling spectral diagnostics, structural fragility analysis, and coherent forward risk assessment. As such, it offers a unified mathematical architecture for studying stability, persistence, and collapse in nonlinear stochastic systems.

Acknowledgments

Mathematics were manually audited by the authors, verified for mathematical accuracy using Wolfram Mathematica 14.3, and refined to ensure alignment with the theoretical model.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

- M. Aloud, E. P. K. Tsang, R. Olsen, and A. Dupuis. A directional-change events approach for studying financial time series. Discussion Paper 2011-28, Economics E-Journal, 2011.
- A. Bakhach, V. Chinthapati, E. P. K. Tsang, and A. E. Sayed. Intelligent dynamic backlash agent: A trading strategy based on the directional change framework. *Algorithms*, 11(11):171, 2018. doi: 10.3390/a11110171.
- M. Berman and R. Schoenfeld. Invariants in experimental data on linear kinetics and the formulation of models. *Journal of Applied Physics*, 27(11):1361–1370, 1956.
- D. G. Covell, M. Berman, and C. DeLisi. Mean residence time—theoretical development, experimental determination, and practical use. *Mathematical Biosciences*, 72(2):213–244, 1984.
- J. Eisenfeld. On mean residence times in compartments. *Mathematical Biosciences*, 57:265–278, 1981. doi: [https://doi.org/10.1016/0025-5564\(81\)90106-1](https://doi.org/10.1016/0025-5564(81)90106-1).

- F. Gao and X. He. Survival analysis: Theory and application in finance. In *Handbook of Financial Econometrics, Statistics, Technology, and Risk Management*, pages 3903–3934. World Scientific, Singapore, 2025.
- J. B. Glattfelder, A. Dupuis, and R. B. Olsen. Patterns in high-frequency fx data: Discovery of 12 empirical scaling laws. *Quantitative Finance*, 11(4):599–614, 2011. doi: 10.1080/14697688.2010.481255.
- J. D. Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57(2):357–384, 1989.
- J. D. Hamilton. Macroeconomic regimes and regime shifts. In *Handbook of Macroeconomics*, volume 2, pages 163–201. Elsevier, Amsterdam, 2016.
- J. Z. Hearon. Residence times in compartmental systems and the moments of a certain distribution. *Mathematical Biosciences*, 15(1):69–77, 1972. ISSN 0025-5564. doi: [https://doi.org/10.1016/0025-5564\(72\)90063-6](https://doi.org/10.1016/0025-5564(72)90063-6).
- J. Z. Hearon. Residence times in compartmental systems with and without inputs. *Mathematical Biosciences*, 55(3):247–257, 1981. ISSN 0025-5564. doi: [https://doi.org/10.1016/0025-5564\(81\)90098-5](https://doi.org/10.1016/0025-5564(81)90098-5).
- L. Himmelmann. *HMM: HMM – Hidden Markov Models*, 2010. URL <https://CRAN.R-project.org/package=HMM>. R package version 1.0.
- J. A. Jacquez. The inverse problem for compartmental systems. *Mathematics and Computers in Simulation*, 24(6):452–459, December 1982. doi: 10.1016/0378-4754(82)90642-5.
- J. A. Jacquez. Identifiability: the first step in parameter estimation. *Federation Proceedings*, 46(8): 2477–2480, July 1987.
- J. A. Jacquez. Identifiability and parameter estimation. *Journal of Parenteral and Enteral Nutrition*, 15(3):55S–59S, May 1991. Supplement.
- J. A. Jacquez. Design of experiments. *Journal of the Franklin Institute*, 335(2):259–279, 1998. doi: 10.1016/S0016-0032(97)00004-5.
- J. A. Jacquez. Density functions of residence times for deterministic and stochastic compartmental systems. *Mathematical Biosciences*, 180(1):127–139, 2002. doi: [https://doi.org/10.1016/S0025-5564\(02\)00110-4](https://doi.org/10.1016/S0025-5564(02)00110-4).

- J. A. Jacquez and P. Greif. Numerical parameter identifiability and estimability: Integrating identifiability, estimability, and optimal sampling design. *Mathematical Biosciences*, 77(1-2): 201–227, December 1985. doi: 10.1016/0025-5564(85)90098-7.
- J. A. Jacquez and T. Perry. Parameter estimation: Local identifiability of parameters. *American Journal of Physiology-Endocrinology and Metabolism*, 258(4):E727–E736, May 1990. doi: 10.1152/ajpendo.1990.258.4.E727.
- J. A. Jacquez and C. P. Simon. Qualitative theory of compartmental systems. *SIAM Review*, 35: 43–79, 1993. doi: 10.1137/1035003.
- D. G. Kleinbaum and M. Klein. *Survival Analysis*. Springer, New York, 3 edition, 2012.
- O. A. Linares, J. A. Jacquez, L. A. Zech, M. J. Smith, J. A. Sanfield, L. A. Morrow, S. G. Rosen, and J. B. Halter. Norepinephrine metabolism in humans. kinetic analysis and model. *The Journal of Clinical Investigation*, 80(5):1332–1341, 11 1987. doi: 10.1172/JCI113210.
- O. A. Linares, L. A. Zech, J. A. Jacquez, S. G. Rosen, J. A. Sanfield, L. A. Morrow, M. A. Supiano, and J. B. Halter. Effect of sodium-restricted diet and posture on norepinephrine kinetics in humans. *American Journal of Physiology-Endocrinology and Metabolism*, 254(2):E222–E230, 1988. doi: 10.1152/ajpendo.1988.254.2.E222. PMID: 3348372.
- L. Oelschläger, T. Adam, and T. Michelot. **fHMM**: Fitting hidden Markov models to financial data. *Journal of Statistical Software*, 104(10):1–25, 2023.
- T. Rydén, T. Teräsvirta, and S. Åsbrink. Stylized facts of daily return series and the hidden markov model. *Journal of Applied Econometrics*, 13(3):217–244, 1998.
- J. A. Sánchez-Espigares and A. López-Moreno. **MSwM**: Fitting markov switching models. *R package version 1.5*, 2021. URL <https://CRAN.R-project.org/package=MSwM>.
- E. P. K. Tsang. Directional changes, definitions. Working Paper WP050-10, Centre for Computational Finance and Economic Agents (CCFEA), University of Essex, 2010.
- E. P. K. Tsang. Directional changes: A new way to look at price dynamics. In *International Conference on Computational Intelligence, Communications, and Business Analytics*, pages 45–55. Springer, 2017.
- E. P. K. Tsang and J. Chen. Regime change detection using directional change indicators in the foreign exchange market to chart brexit. *IEEE Transactions in Emerging Topics in Computational Intelligence*, 2(3):185–193, 2018. doi: 10.1109/TETCI.2017.2769057.

- E. P. K. Tsang, R. Tao, A. Serguieva, and S. Ma. Profiling high-frequency equity price movements in directional changes. *Quantitative Finance*, 17(2):217–225, 2017. doi: 10.1080/14697688.2016.1211798.
- I. Visser and M. Speekenbrink. **depmixS4**: An R package for hidden mixture models. *Journal of Statistical Software*, 36(7):1–21, 2010.

Copyright: ©2026 Oscar Linares, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.