Koopman Invariants as Drivers of Emergent Time-Series Clustering in Joint-Embedding Predictive Architectures

Pablo Ruiz-Morales[1,3], **Dries Vanoost**[2,3], **Davy Pissoort**[2,3], **Mathias Verbeke**[1,3]
[1]Declarative Languages and Artificial Intelligence (DTAI), M-Group, KU Leuven, Bruges, Belgium
[2]ESAT-WaveCore, M-Group, KU Leuven, Bruges, Belgium
[3]Flanders Make@KU Leuven, Belgium
{pablo.ruizmorales, dries.vanoost, davy.pissoort, mathias.verbeke}@kuleuven.be

Abstract

Joint-Embedding Predictive Architectures (JEPAs), a powerful class of self-supervised models, exhibit an unexplained ability to cluster time-series data by their underlying dynamical regimes. We propose a novel theoretical explanation for this phenomenon, hypothesizing that JEPA's predictive objective implicitly drives it to learn the invariant subspace of the system's Koopman operator. We prove that an idealized JEPA loss is minimized when the encoder represents the system's regime indicator functions, which are Koopman eigenfunctions. This theory was validated on synthetic data with known dynamics, demonstrating that constraining the JEPA's linear predictor to be a near-identity operator is the key inductive bias that forces the encoder to learn these invariants. We further discuss that this constraint is critical for selecting this interpretable solution from a class of mathematically equivalent but entangled optima, revealing the predictor's role in representation disentanglement. This work demystifies a key behavior of JEPAs, provides a principled connection between modern self-supervised learning and dynamical systems theory, and informs the design of more robust and interpretable time-series models.

Extended version — https://arxiv.org/abs/2511.09783

1 Introduction

Self-Supervised Learning (SSL) has emerged as a powerful paradigm for learning rich data representations from unlabeled data, driving significant progress across diverse domains (Chen et al. 2020; Grill et al. 2020; Chen and He 2021). These methods learn by solving pretext tasks, with the hope that the learned representations capture underlying structures of the data. However, despite their empirical power, the precise mechanisms by which some SSL architectures discover these structures often remain opaque, treating critical model components as black boxes. This underscores a pressing need for theoretical frameworks that can provide a principled understanding of their behavior and guide future development.

Among the diverse SSL strategies, Joint-Embedding Predictive Architectures (JEPAs) (LeCun 2022; Assran et al. 2023) offer a compelling approach. Rather than reconstructing raw inputs, JEPA is a non-generative approach that learns by predicting future data representations within an abstract, jointly-embedded latent space. This focus on abstract predictability has proven highly effective across diverse domains, spanning from static images to dynamic video and time-series data (Verdenius, Zerio, and Wang 2024; Assran et al. 2023; Bardes et al. 2024).

Intriguingly, when applied to time-series data, JEPA models often yield latent embeddings that spontaneously cluster by underlying, unannotated dynamical regimes. This behavior is not universally observed in other representation learning frameworks, even those employing encoder architectures of similar capacity but with different objectives (e.g., reconstruction-based autoencoders). For instance, as empirically demonstrated in Section 5 (Figure 3), JEPA can effectively disentangle distinct dynamical modes where a comparable autoencoder (AE) fails to do so.

This pronounced difference in latent organization immediately poses a crucial question: Why does JEPA's predictive objective lead to this regime-aware clustering? What intrinsic mechanism within JEPA drives this emergence of order from apparently unstructured input? Addressing this question is vital not only for understanding JEPA itself but also for developing more principled and effective SSL methods.

To address this, we turn to dynamical systems theory, specifically the Koopman operator framework (Koopman 1931; Mezić 2005). The Koopman operator offers a powerful way to analyze nonlinear dynamical systems by lifting observations into a space where their evolution becomes linear. This approach has inspired various machine learning techniques aiming to learn these linear representations. For example, significant research has focused on using deep autoencoders to learn intrinsic coordinates where dynamics evolve linearly under a learned Koopman operator, often with auxiliary networks to enforce desired properties like parsimony or to capture continuous spectra (Lusch, Kutz, and Brunton 2018; Yeung, Kundu, and Hodas 2019). Architectures like Linearly Recurrent Autoencoder Networks explicitly constrain latent dynamics to follow a linear recurrent layer, effectively learning finite-dimensional Koopman approximations (Azencot et al. 2020; Otto and Rowley 2019). This philosophy of enforcing linear latent dynamics has achieved remarkable success in modern structured State-Space Models (SSMs) like Mamba (Gu, Goel, and Ré 2022; Gu and Dao 2024), which represent the current state-

of-the-art for many sequence modeling tasks. These methods, building upon foundations like Dynamic Mode Decomposition and its extensions (Schmid 2010; Korda and Mezić 2018), demonstrate the utility of Koopman theory for system identification and representation learning where linear evolution is a primary target.

Other approaches, such as VAMPnets (Mardt et al. 2018) and Time-Lagged Autoencoders (Wehmeyer and Noé 2018), are designed to learn slow collective variables or eigenfunctions of the underlying system's transfer operator, which are crucial for understanding long-timescale dynamics and regime transitions. A distinct but related goal in representation learning is to achieve disentanglement, where latent variables are explicitly regularized to capture independent factors of variation in the data (Higgins et al. 2017; Kim and Mnih 2018). While these methods successfully identify key dynamical or generative modes, they often employ specialized objectives directly tied to these targets.

In contrast to these approaches that explicitly model system dynamics, optimize for specific eigenfunctions, or regularize for disentangled factors, the mechanism within JEPA is implicit. While many self-supervised methods for time series rely on contrastive objectives that learn similarity based on temporal proximity (Yue et al. 2022), we hypothesize that JEPA's purely predictive objective in a latent space is what drives it to learn functions that are invariant under the system's evolution within distinct dynamical regimes.

We will demonstrate that these invariant functions correspond to the indicator functions of these regimes, which are, in fact, eigenfunctions of the system's Δ-step Koopman operator associated with a unit-magnitude eigenvalue. Thus, while not designed as a Koopman modeling tool, JEPA's core learning principle appears to converge on identifying these fundamental Koopman invariants when distinct, stable dynamical regimes are present.

Our contributions are twofold:

1. We provide a theoretical derivation showing that an idealized JEPA loss function is minimized when its encoder learns to span the space of these Koopman-invariant regime indicators. This offers a first-principles explanation for the observed clustering.
2. We detail an empirical validation strategy using synthetic time-series data with known underlying regimes. Beyond standard t-SNE visualization, our methodology includes a novel set of analyses focused on the learned linear predictor matrix M by examining its Frobenius norm difference from identity, symmetry, eigenvalue spectrum, and action on empirically derived cluster centroids to directly test the theoretical prediction that M behaves as an identity operator on the learned regime subspace.

By clarifying the mechanism behind JEPA's emergent clustering, this work not only offers a deeper understanding of this powerful SSL architecture but also strengthens the promising connections between modern deep learning and the established mathematics of dynamical systems.

This paper is structured as follows: Section 2 introduces JEPA and Koopman operator theory in more detail, as the foundation for the theoretical derivation in Section 3. Section 4 details the experimental setup for validation, followed by the obtained results and discussion in Section 5. Finally, in Section 6, we synthesize our findings and outline promising directions for future research.

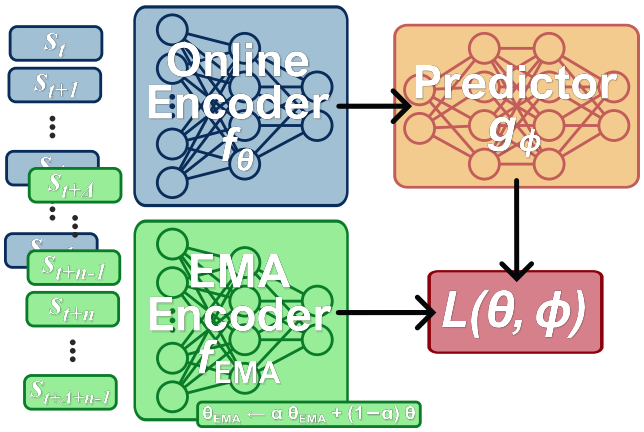

Figure 1: Conceptual schematic of the Joint-Embedding Predictive Architecture (JEPA). The online encoder and predictor aim to predict the representation of a future target window as generated by the slowly evolving EMA encoder.

2 Theoretical Framework

To understand the emergent clustering phenomenon in JEPAs, we first need to formalize its learning process and then introduce the mathematical tools from dynamical systems theory that will allow us to analyze its behavior. Our central aim is to connect JEPA's predictive objective to the discovery of underlying dynamical regimes.

JEPA Model and Idealized Loss

We consider time-series windows $x_t = (s_t, \ldots, s_{t+n-1}) \in \mathbb{R}^d \equiv \mathcal{X}$, drawn from a stationary process with invariant measure μ. The JEPA model consists of three neural network components:

- An online encoder $f_\theta : \mathcal{X} \to \mathbb{R}^k$, which maps an input window x_t to a latent representation $z_t = f_\theta(x_t)$.
- An online predictor $g_\phi : \mathbb{R}^k \to \mathbb{R}^k$, which takes z_t and predicts the latent representation of a future window $x_{t+\Delta}$ (where $\Delta \geq 1$ is the prediction horizon).
- A target encoder $f_{\text{EMA}} : \mathcal{X} \to \mathbb{R}^k$, structurally identical to f_θ, whose parameters θ_{EMA} are updated as an Exponential Moving Average (EMA) of the online encoder's parameters: $\theta_{EMA} \leftarrow \alpha\,\theta_{EMA} + (1-\alpha)\,\theta$.

The momentum coefficient α is typically chosen to be very close to 1 (e.g., 0.996 to 0.999), ensuring that θ_{EMA} represents a slowly evolving, stable average of θ (Grill et al. 2020; He et al. 2020). The target encoder provides stable targets $z_{t+\Delta} = f_{\text{EMA}}(x_{t+\Delta})$ for training.

The online network parameters θ and ϕ are learned by minimizing the predictive loss, typically a squared Euclidean distance, averaged over the data distribution μ of the

input windows:

$$L(\theta, \phi) = \mathbb{E}_{x_t \sim \mu} \left[\|g_\phi(f_\theta(x_t)) - f_{\text{EMA}}(x_{t+\Delta})\|_2^2 \right] \quad (1)$$

To perform a tractable theoretical analysis of the representations f_θ JEPA learns, we make a key simplifying assumption. We assume that the target encoder f_{EMA} closely tracks the online encoder f_θ. Specifically, if the rate of change of the online parameters θ per training step is small relative to $(1 - \alpha)$, then the difference $\|\theta - \theta_{EMA}\|$ remains small. Consequently, for well-behaved neural network functions f, this implies $f_{\text{EMA}}(y) \approx f_\theta(y)$ for any relevant input y. This approximation is common in the analysis of similar self-supervised methods employing EMA targets (Grill et al. 2020; He et al. 2020; Tian, Chen, and Ganguli 2021) and is implicitly validated by their empirical success, which relies on effective target stabilization.

Under this condition of close tracking, the objective function in (1) can be approximated by:

$$L(f, g) \approx \mathbb{E}_{x_t \sim \mu} \left[\|g(f(x_t)) - f(x_{t+\Delta})\|_2^2 \right], \quad (2)$$

where $f \equiv f_\theta$ and $g \equiv g_\phi$. Our theoretical development will be based on this idealized loss, while acknowledging that the EMA dynamics introduce a slight deviation in practice.

Koopman Operator

To analyze the functions f that JEPA might learn, particularly in the context of time-series data exhibiting underlying dynamical regimes, we employ the Koopman operator framework. We view the sequence of windows $\{x_t\}$ as states of a discrete-time dynamical system on $\mathcal{X}$, evolving under the invariant measure μ.

The Koopman operator provides a way to study the evolution of functions (observables) $\psi : \mathcal{X} \to \mathbb{C}$ defined on the state space. We focus on observables in the Hilbert space $L^2(\mathcal{X}, \mu)$.

Definition 2.1 (Δ-Step Koopman Operator)**.** The Δ-step Koopman operator $\mathcal{K} \equiv \mathcal{K}_{(\Delta)} : L^2(\mathcal{X}, \mu) \to L^2(\mathcal{X}, \mu)$ transforms an observable ψ into its conditional expectation Δ steps into the future:

$$(\mathcal{K}\psi)(x) = \mathbb{E}[\psi(x_{t+\Delta}) \mid x_t = x]. \quad (3)$$

A fundamental property of $\mathcal{K}$ is its linearity, regardless of the non-linearity of the underlying system generating x_t. It is also a contraction on $L^2(\mathcal{X}, \mu)$. Of particular interest are its eigenfunctions ψ_j, which satisfy $\mathcal{K}\psi_j = \nu_j \psi_j$ for eigenvalues $\nu_j \in \mathbb{C}$. Eigenfunctions with $\nu_j = 1$ represent quantities that are invariant in expectation under the Δ-step dynamics. The components of JEPA's learned encoder $f_\theta(x_t)$ can be seen as a vector of observables, and understanding their relationship with the Koopman operator is key to our analysis.

3 Theoretical Justification for Clustering

Having established the JEPA learning paradigm and the Koopman operator framework, we now develop our central theoretical argument. We will demonstrate that under specific assumptions about the data dynamics, (2) incentivizes the encoder f to learn representations that correspond to underlying, discrete dynamical regimes. This occurs because these regime-specific representations are characterized by functions that are invariant under the Koopman operator, making them optimally predictable.

Dynamical Regimes and Invariant Observables

The cornerstone of our argument is the assumption that the observed time-series data, while potentially complex, arises from a system that operates in a finite number of distinct dynamical modes or regimes.

Assumption 3.1 (Finite Mixture of Ergodic Regimes)**.** *The invariant measure μ governing the time-series windows $x_t \in \mathcal{X}$ decomposes as a finite convex mixture of r distinct ergodic component measures:*

$$\mu = \sum_{i=1}^{r} \alpha_i \mu_i, \text{ where } r \in \mathbb{N},\ \alpha_i > 0, \sum_{i=1}^{r} \alpha_i = 1 \quad (4)$$

Each μ_i is an ergodic invariant measure supported on a measurable set $\mathcal{X}_i \subset \mathcal{X}$. These supports $\{\mathcal{X}_i\}_{i=1}^r$ are essentially disjoint, and trajectories starting in $\mathcal{X}_i$ remain confined to $\mathcal{X}_i$ for all future times (dynamical immiscibility).

This assumption allows us to define regime indicator functions $\chi_i(x) := \mathbf{1}_{\mathcal{X}_i}(x)$. These functions are linearly independent and span an r-dimensional subspace $\mathcal{V} := \text{span}\{\chi_1, \ldots, \chi_r\}$.

Lemma 3.2 (Properties of Regime Indicators and $\mathcal{V}$)**.** *Under Assumption 3.1:*

(a) Each regime indicator χ_i is an eigenfunction of $\mathcal{K}$ with eigenvalue 1: $\mathcal{K}\chi_i = \chi_i$

(b) Each χ_i is pathwise invariant over Δ steps: $\chi_i(x_{t+\Delta}) = \chi_i(x_t)$

(c) Consequently, any function $\psi \in \mathcal{V}$ satisfies $\mathcal{K}\psi = \psi$ and $\psi(x_{t+\Delta}) = \psi(x_t)$

(d) The subspace $\mathcal{V}$ is precisely the eigenspace of $\mathcal{K}$ corresponding to the eigenvalue 1, and its dimension is r.

Proof Intuition. Properties (a) and (b) stem directly from the dynamical immiscibility of regimes: if a system is in regime $\mathcal{X}_i$, it is expected to remain there, and its indicator χ_i will deterministically remain 1. Property (c) follows by linearity. Property (d) is a fundamental result from ergodic decomposition theory, linking the number of ergodic components to the dimension of the invariant subspace of the Koopman operator. The detailed proofs are provided in Appendix A. □

Lemma 3.2(c) is critical: functions in $\mathcal{V}$ are not just invariant in expectation but are also perfectly predictable on a pathwise basis with respect to the regimes, i.e., their future value $f(x_{t+\Delta})$ is identical to their current value $f(x_t)$. This makes them prime candidates for what JEPA might learn.

JEPA Loss Minimization and Koopman Invariants

We now connect the idealized loss (2) to the learning of these Koopman-invariant functions. For analytical clarity, especially in isolating the encoder's role in finding predictable structures, we introduce an assumption about the predictor.

Assumption 3.3 (Linear Predictor)**.** *The predictor $g : \mathbb{R}^k \to \mathbb{R}^k$ is a linear transformation, i.e., $g(z) = Mz$ for some matrix $M \in \mathbb{R}^{k\times k}$ with learnable parameters.*

Under this assumption, let $f(x) = \vec{\psi}(x)$ be the vector of k observables learned by the encoder. The idealized JEPA loss (2) becomes $L(f, M) = \mathbb{E}_{x\sim\mu}\big[\|M\vec{\psi}(x) - \vec{\psi}(x_{t+\Delta})\|_2^2\big]$. This loss can be decomposed using the Koopman operator $\mathcal{K}$, as derived in Appendix A:

$$L(f, M) = \underbrace{\mathbb{E}_x\big[\|M\vec{\psi}(x) - (\mathcal{K}\vec{\psi})(x)\|_2^2\big]}_{\text{Term 1: Mean Prediction Error}} + \underbrace{\mathbb{E}_x\big[\|(\mathcal{K}\vec{\psi})(x) - \vec{\psi}(x_{t+\Delta})\|_2^2\big]}_{\text{Term 2: Inherent Stochasticity Error}} \tag{5}$$

JEPA aims to minimize $L(f, M)$ by jointly optimizing f (i.e., $\vec{\psi}$) and M. The loss is zero if and only if both Term 1 and Term 2 are zero. Term 2 quantifies the degree to which the learned observables $\vec{\psi}(x)$ deviate from their conditional expectation $(\mathcal{K}\vec{\psi})(x)$ along actual trajectories. Term 1 quantifies how well the linear predictor $M\vec{\psi}(x)$ can match this conditional expectation.

Our theory shows that functions spanning the regime-invariant subspace $\mathcal{V}$ are optimal for minimizing this loss.

Theorem 3.4 (JEPA Learns Regime Indicators)**.** *Let Assumptions 3.1 and 3.3 hold. Assume the encoder has sufficient capacity, i.e., its latent dimension $k \geq r$. The JEPA loss* (5) *achieves its global minimum if and only if:*

(a) The components f_j of the encoder output are such that $(\mathcal{K}f_j)(x) = f_j(x_{t+\Delta})$ for μ almost everywhere (a.e.).
(b) The predictor matrix M satisfies $Mf(x) = (\mathcal{K}f)(x)$ for μ a.e. x.

These conditions are simultaneously satisfied if the components $f_j(x)$ of the encoder $f(x)$ belong to the invariant subspace $\mathcal{V}$, and the predictor matrix M acts as the identity transformation on the subspace of $\mathbb{R}^k$ spanned by $f(\mathcal{X})$.

Specifically, if $f^*(x) = (\chi_1(x), \ldots, \chi_r(x), \vec{0}_{k-r})^T$, then $L(f^*, M)$ is minimized by any M^* whose action on the subspace spanned by the non-zero components of $f^*(\mathcal{X})$ is identity and zero elsewhere.

Proof Intuition. If each component $f_j \in \mathcal{V}$, then by Lemma 3.2(c), we have both $(\mathcal{K}f_j)(x) = f_j(x)$ and $f_j(x_{t+\Delta}) = f_j(x_t)$.

Thus, if the encoder learns functions f_j that are (or span) the regime indicators in $\mathcal{V}$, both terms of the loss can be driven to zero with a predictor M that effectively acts as an identity map on these learned, invariant representations.

For the specific f^* given, M^* being identity on the first r components achieves this. Conversely, for the loss to be zero, Term 2 requires $(\mathcal{K}f_j)(x_t) = f_j(x_{t+\Delta})$ almost surely (a.s.). If M then ensures Term 1 is zero by $Mf_j = \mathcal{K}f_j$, and if M is to be simple (e.g., identity-like for these f_j), it implies $\mathcal{K}f_j = f_j$, pushing f_j towards $\mathcal{V}$. The full details are provided in Appendix A. □

Implication for Latent Space Clustering

Theorem 3.4 provides a direct explanation for the empirically observed clustering. If the JEPA encoder $f(x)$, with latent dimension $k \geq r$, learns representations whose components f_j span the r-dimensional invariant subspace $\mathcal{V}$, then $f(x)$ effectively becomes a representation of the regime indicators.

For instance, if $f(x)$ is an invertible linear transformation of the vector $\vec{\chi}(x) = (\chi_1(x), \ldots, \chi_r(x))^T$, say $f(x) = A\vec{\chi}(x)$ for a $k \times r$ matrix A of rank r. When a window x belongs to regime $\mathcal{X}_i$, $\vec{\chi}(x)$ becomes e_i (the i-th standard basis vector in $\mathbb{R}^r$). Consequently, the encoder output is $f(x) = Ae_i$, which is simply the i-th column of matrix A.

Therefore, all input windows x_t originating from the same dynamical regime $\mathcal{X}_i$ are mapped by the encoder f to the same space (or a very tight region, allowing for approximation errors and noise) Ae_i in the latent space R^k. Since A has rank r, the r vectors $\{Ae_i\}_{i=1}^r$ are distinct (or linearly independent if $k \geq r$). This mapping naturally results in r distinct clusters in the latent space, with each cluster corresponding precisely to one of the underlying regimes.

While our core theoretical result relies on the assumption of a linear predictor $g(z) = Mz$ for analytical tractability, the general intuition extends to non-linear predictors. A sufficiently expressive non-linear predictor g_ϕ would aim to approximate the conditional expectation $g_\phi(f(x_t)) \approx \mathbb{E}[f(x_{t+\Delta}) \mid f(x_t)]$. If the encoder f learns representations $z_t = f(x_t)$ that are elements of $\mathcal{V}$, then $z_{t+\Delta} = z_t$ a.s. In this scenario, the conditional expectation $\mathbb{E}[z_{t+\Delta} \mid z_t] = z_t$. Thus, an optimal non-linear predictor would learn to approximate an identity map, $g_\phi(z_t) \approx z_t$, for these highly predictable, regime-specific representations. The fundamental drive to find representations $f(x_t)$ for which $f(x_{t+\Delta})$ is "simply" predictable from $f(x_t)$ remains, and functions in $\mathcal{V}$ (for which $f(x_{t+\Delta}) = f(x_t)$) represent the epitome of such simplicity.

4 Experimental Validation

To rigorously test the predictions of our Koopman-based theory, we employ a novel empirical strategy. Using a synthetic dataset with known underlying regimes, we perform a series of targeted, quantitative analyses on the learned linear predictor, M, to verify its predicted behavior as an identity operator on the learned invariant subspace.

Synthetic Dataset Generation

To create an environment where Assumption 3.1 (Finite Mixture of Ergodic Regimes) is satisfied by construction, we generate a synthetic dataset comprising $r = 18$ distinct dynamical regimes. The objective is to provide the JEPA model with input data that clearly embodies the structured, multimodal dynamics our theory addresses. Each regime is designed to exhibit unique temporal characteristics, ensuring clear distinctions and facilitating the analysis of JEPA's ability to differentiate them.

Master sequences, each of length $L_{master} = 1024$ time steps, are generated for every regime. For these experiments, additive observation noise is set to zero for all deterministic

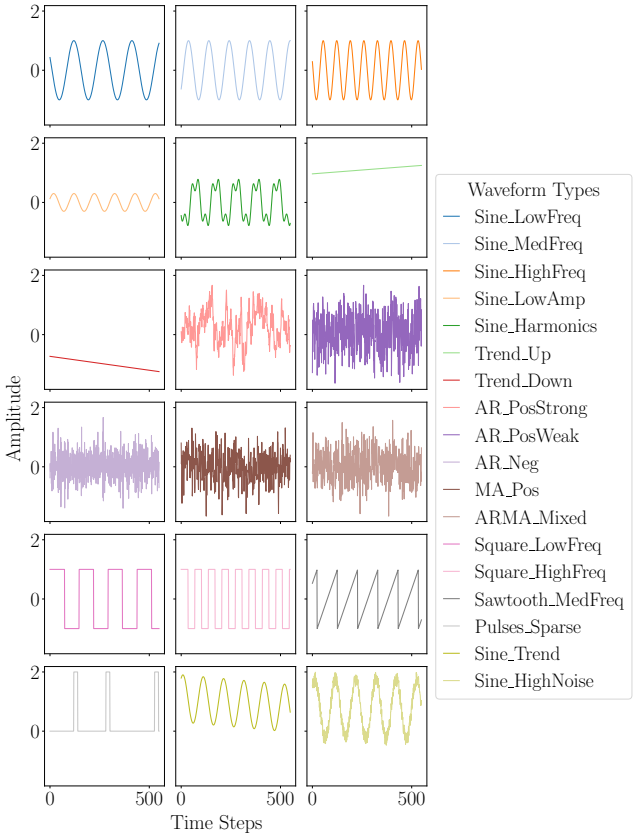

Figure 2: Example waveforms illustrating the diversity of the synthetic dataset.

signal components, ensuring the cleanest possible testbed to verify our theoretical claims without the confounding effects of stochastic observation noise. Stochastic processes, such as Autoregressive Moving Average (ARMA) models, naturally incorporate their own intrinsic process noise.

The repertoire of $r = 18$ regimes encompasses a diverse set of dynamics crucial for testing the robustness of our theory: Periodic signals include several sinusoidal variations differing in frequency, amplitude, and harmonics, with phases randomized per sequence. Square waves and a sawtooth wave provide examples of non-smooth periodicities.

To model stochastic dynamics, we include Autoregressive (AR) models with varying dependency coefficients, a Moving Average (MA) process, and a mixed ARMA model.

Aperiodic and event-based signals are represented by linear trends with both positive and negative slopes, featuring per-sequence randomization of slope and intercept, and sequences with sparse, randomly located positive pulses. Finally, to explore more complex interactions, the dataset includes combined signals like a sinusoid superimposed on a linear trend, and a high-noise variant consisting of a sinusoid with significantly increased internal process noise. Examples of each type of waveform are depicted in Figure 2, and a comprehensive list of parameters for each generative process is detailed in Appendix B.

From each master sequence, we extract a single context-target pair. The context window, x_t, consists of the initial $n_c = 768$ steps. The target window, $x_{t+\Delta}$, is defined by shifting forward by a prediction horizon of $\Delta = 256$ steps, also taking a 768-step window.

This windowing scheme creates a substantial overlap: the steps from s_Δ to s_{n_c-1} are present in both the context and target. This design serves a dual purpose: it tasks the model with maintaining representational consistency for the overlapping data, while also requiring it to predict the representation of the novel future segment (s_{n_c} to $s_{\Delta+n_c-1}$). We note that experiments with a completely non-overlapping windowing scheme ($\Delta \geq n_c$) also yielded satisfactory results.

A total of 10,000 sequences are generated for each of the $r = 18$ regimes, yielding 180,000 context-target pairs. This dataset is deterministically partitioned at the sequence level into training (70%), validation (20%), and test (10%) sets, with all regime types proportionally represented across splits. Prior to window extraction, each master sequence undergoes per-sequence standardization to encourage the model to learn shape-based and relative dynamical features.

JEPA Model Configuration

To instantiate the JEPA model whose idealized behavior was analyzed in Section 3, and to empirically test our theoretical predictions, we configure the encoder and predictor architectures with specific considerations for this study. All models are implemented in PyTorch.

The encoder f_θ is a one-dimensional Convolutional Neural Network designed to process the input time-series windows x_t. Its architecture consists of four convolutional layers, each followed by ReLU activation. The output of the convolutional blocks is flattened and passed through a linear projection head to produce a latent representation $z_t \in \mathbb{R}^k$. The latent dimension k was set to 32, ensuring $k \geq r$, providing sufficient capacity for the encoder to potentially represent the r distinct regimes as predicted by our theory.

A key component for directly testing the predictions regarding the predictor's behavior (Theorem 3.4) involves using a linear predictor for g_ϕ. This predictor implements a linear transformation $g(z) = Mz$, where $M \in \mathbb{R}^{k\times k}$, and was configured without bias to simplify analysis. To specifically probe for the existence and stability of the theoretically optimal identity-like solution, our primary analysis stems from experiments where M was initialized as an identity matrix. To explore the broader optimization landscape under more standard conditions, separate control experiments were conducted where M was initialized using a standard random scheme. This dual approach allows us to both verify the existence of an interpretable solution and assess its uniqueness.

For comparison, and to observe clustering under more typical JEPA conditions, we also configure experiments using a standard non-linear multi-layer perceptron (MLP) predictor. This predictor consists of two hidden layers with ReLU activations, where the hidden dimension is two times the latent dimension k. The full details of the model are provided in Appendix C.

The target encoder f_{EMA} is a direct copy of the online encoder f_θ, with its parameters updated via an Exponential

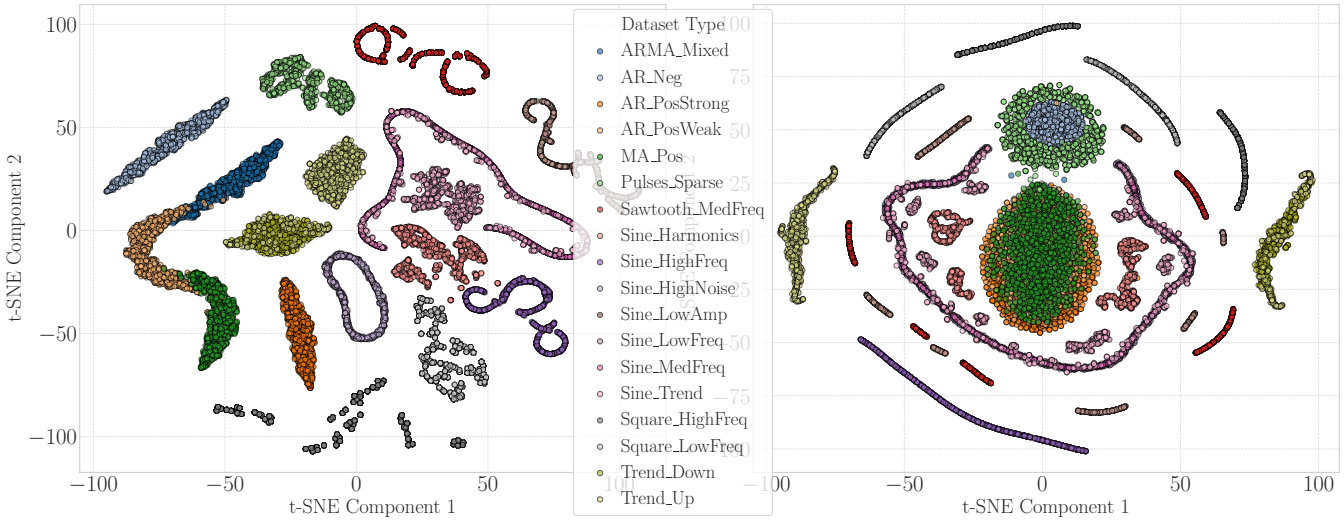

Figure 3: Latent space visualization (t-SNE) of test set embeddings. Left: JEPA embeddings often form distinct clusters corresponding to underlying dynamical regimes (indicated by colors). Right: Embeddings from a conventional autoencoder with an identical encoder architecture may not exhibit such clear regime-based separation on the same data.

Moving Average (EMA) using a decay rate of $\alpha = 0.996$.

5 Results and Discussion

Our analysis focuses on verifying the key predictions derived from our Koopman operator-based theory: namely, the emergence of regime-aligned clustering in the latent space and the characteristic behavior of the learned predictor, particularly in the idealized linear case. All reported results are from evaluations on the held-out test set.

A primary outcome of our theory is that JEPA's encoder f_θ should learn representations $z_t = f_\theta(x_t)$ that distinguish the underlying dynamical regimes, leading to distinct clusters in the latent space. To investigate this, we projected the k-dimensional latent embeddings of test set windows into a 2D space using t-SNE (van der Maaten and Hinton 2008). The points were then color-coded according to their ground-truth regime labels.

Figure 3 presents these visualizations. As hypothesized, the JEPA model trained with a standard non-linear MLP predictor demonstrates a clear formation of clusters that align strongly with the ground-truth dynamical regimes. We quantified this alignment using K-Means clustering (K $= 18$), which revealed a mean cluster purity of 65.48% for JEPA's embeddings. This emergent order supports our argument that the predictive objective learns features separable by the underlying data-generating modes.

In contrast, a conventional autoencoder with an identical encoder architecture fails to separate the dynamical regimes for the same data, achieving a mean cluster purity of only 38.81%. This difference highlights the efficacy of JEPA's abstract predictive objective over a purely reconstructive one for uncovering and organizing representations by their underlying dynamical structure.

Analysis of the Learned Linear Predictor Matrix

Our theory (Theorem 3.4), under the assumption of a linear predictor $g(z) = Mz$, implies that if the encoder learns regime indicator functions (elements of $\mathcal{V}$), then M should act as an identity transformation on the subspace spanned by these learned representations. We investigate this by analyzing the $k \times k$ matrix M learned by the JEPA model equipped with a linear predictor (initialized as $M \approx I_k$).

The learned matrix M from our identity-initialized experiment converged to a near-perfect identity transformation. This was confirmed through three key quantitative properties. First, its deviation from the identity matrix I_k was minimal, with a relative Frobenius error ($||M - I_k||_F / ||M||_F$) of just 2.34%. Second, the matrix was highly symmetric, another crucial property of an identity operator, with its skew-symmetric relative norm ($||M - M^T||_F / ||M||_F$) measuring only 2.06%. Finally, and most critically, its eigenvalue spectrum, displayed in Figure 4, reveals the mechanism behind this behavior: it is dominated by r eigenvalues near 1.0, indicating that M has learned to preserve the r-dimensional subspace of the learned regimes while potentially attenuating all other, less predictable dimensions.

To further test M as an identity operator on the representations of the learned regimes, we identified $r = 18$ cluster centroids $\{c_i\}_{i=1}^{18}$ from the test set embeddings using K-Means. We then computed the relative Euclidean norm of the difference $\|Mc_i - c_i\|_2 / \|c_i\|_2$ for each centroid. The error was consistently small, averaging just 0.80% across all centroids, as shown in Figure 5. This confirms that M preserves the locations of the regime centroids, aligning with the theoretical prediction that $Mf(x) \approx f(x)$ for $f(x) \in \mathcal{V}$.

Crucially, this interpretable solution is not unique. Our control experiments with a randomly initialized predictor M converged to the same low loss but yielded a dense, non-identity transformation, while also exhibiting clear visual

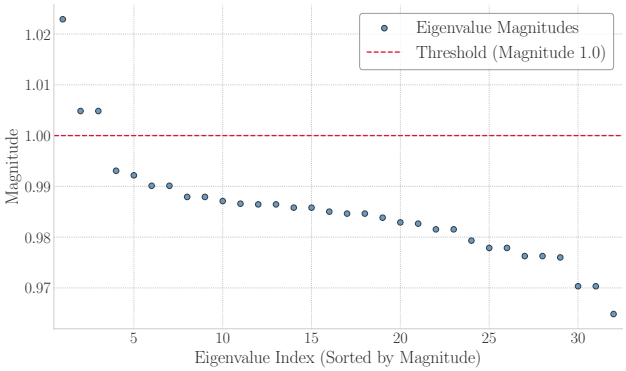

Figure 4: Sorted magnitudes of the eigenvalues of M, showing dominant eigenvalues near 1.0.

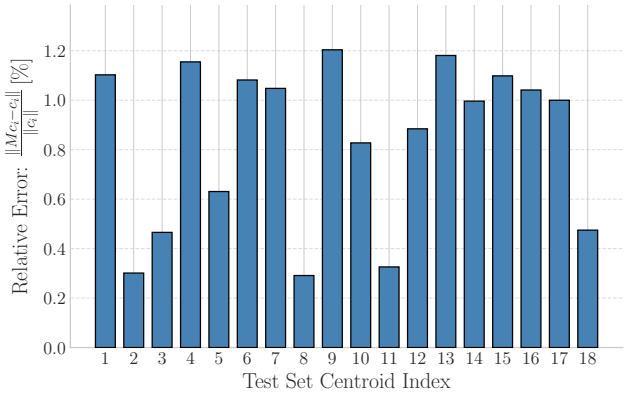

Figure 5: Action of the learned linear predictor M on regime cluster centroids c_i. The small relative error indicates that M largely preserves these regime-representative vectors.

clustering. This finding does not contradict our theory but instead highlights its core implication: the JEPA loss is invariant to any invertible linear transformation (a change of basis) applied to the latent space. While a random initialization finds an equally valid but entangled basis for the optimal subspace, our identity-initialized experiment proves that a canonical, interpretable solution exists and is a stable optimum. This demonstrates that while JEPA's objective successfully identifies the correct invariant subspace, an inductive bias, such as initializing the predictor toward identity, guides the model to a human-interpretable representation.

The empirical results strongly align with the theoretical framework from Section 3. JEPA encoders form latent space clusters that directly correspond to the r ground-truth dynamical regimes (Figure 3), a stark contrast to a standard autoencoder with an identical architecture. This demonstrates that JEPA's predictive objective, not just its capacity, is responsible for uncovering the underlying dynamical structure.

The analysis of the linear predictor M further supports our Koopman-based theory. Its dominant eigenvalues near 1.0 and its near-identity action on cluster centroids both confirm that the predictor learns to preserve the representations of the stable regimes. This is precisely the behavior expected if the encoder f_θ successfully learns functions spanning the invariant subspace $\mathcal{V}$, whose elements $\psi \in \mathcal{V}$ satisfy both $\mathcal{K}\psi = \psi$ and $\psi(x_{t+\Delta}) = \psi(x_t)$ (Lemma 3.2).

Several limitations of the current study should be acknowledged. Our empirical validation relies on synthetic data with well-defined, immiscible regimes that perfectly fit the finite mixture of ergodic regimes model of Assumption 3.1. Real-world time series, however, often feature more complex phenomena, such as gradual transitions or hierarchical structures. Investigating JEPA's behavior under these conditions is an important next step.

On the theoretical side, our analysis made two key idealizations. We utilized a linear predictor for analytical tractability and approximated the EMA dynamics with $f_{\text{EMA}} \approx f_\theta$. While our results are strong under these conditions, a more formal treatment of non-linear predictors and a deeper analysis of the EMA's stabilizing role would provide a more complete understanding of its role in stabilizing the learning of these invariants.

6 Conclusion

This paper has addressed why Joint-Embedding Predictive Architectures often exhibit emergent clustering of their latent representations according to underlying dynamical regimes in time-series data. We proposed a novel theoretical explanation rooted in Koopman operator theory, hypothesizing that JEPA's objective incentivizes its encoder to learn regime indicator functions. These indicators are invariant eigenfunctions (eigenvalue 1) of the Δ-step Koopman operator and characterize the distinct, dynamically immiscible modes of behavior assumed to be present in the data.

Our theoretical derivation, under idealized conditions including a finite mixture of ergodic regimes and a linear predictor, proves that the JEPA loss is minimized when the encoder learns to span this space of indicators. This provides a first-principles basis for the observed clustering, where inputs from the same regime map to common latent locations. Our empirical results on synthetic data strongly support these predictions, showing clear clustering and the predicted behavior of the matrix M.

This work contributes to a deeper principled understanding of JEPA mechanisms grounded in dynamical systems theory. These insights can inspire new SSL objectives that explicitly leverage Koopman theory for more robust and interpretable representation learning. This could lead to improved unsupervised tools for system decomposition, regime identification, and anomaly detection, where deviations from learned regime clusters would signify anomalies.

Future work includes validating these findings on real-world datasets, comparing JEPA's implicit discovery to explicit identification methods, and exploring whether the architecture can learn other Koopman eigenfunctions corresponding to different dynamical modes, such as oscillations or slow decay.

Acknowledgments

Funded by the European Union (grant no. 101168880). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. The European Union can not be held responsible for them. Project website: https://dn-isense.eu/.

For the purpose of open access (OA), as required by Horizon Europe (HE), the author has applied a CC BY public copyright license to the Author Accepted Manuscript (AAM) version resulting from this submission.

References

Assran, M.; Duval, Q.; Misra, I.; Bojanowski, P.; Vincent, P.; Rabbat, M.; LeCun, Y.; and Ballas, N. 2023. Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 15619–15629.

Azencot, O.; Erichson, N. B.; Lin, V.; and Mahoney, M. 2020. Forecasting Sequential Data Using Consistent Koopman Autoencoders. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 475–485.

Bardes, A.; Garrido, Q.; Ponce, J.; Chen, X.; Rabbat, M.; LeCun, Y.; Assran, M.; and Ballas, N. 2024. Revisiting Feature Prediction for Learning Visual Representations from Video. *Transactions on Machine Learning Research*.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*.

Chen, X.; and He, K. 2021. Exploring Simple Siamese Representation Learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 15745–15753.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P. H.; Buchatskaya, E.; Doersch, C.; Pires, B. A.; Guo, Z. D.; Azar, M. G.; Piot, B.; Kavukcuoglu, K.; Munos, R.; and Valko, M. 2020. Bootstrap your own latent a new approach to self-supervised learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*. ISBN 9781713829546.

Gu, A.; and Dao, T. 2024. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. arXiv:2312.00752.

Gu, A.; Goel, K.; and Ré, C. 2022. Efficiently Modeling Long Sequences with Structured State Spaces. arXiv:2111.00396.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9726–9735.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *International Conference on Learning Representations*.

Kim, H.; and Mnih, A. 2018. Disentangling by Factorising. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 2649–2658. PMLR.

Koopman, B. O. 1931. Hamiltonian Systems and Transformation in Hilbert Space. *Proceedings of the National Academy of Sciences*, 17(5): 315–318.

Korda, M.; and Mezić, I. 2018. On convergence of extended dynamic mode decomposition to the Koopman operator. *J. Nonlinear Sci.*, 28(2): 687–710.

LeCun, Y. 2022. A Path Towards Autonomous Machine Intelligence Version 0.9.2, 2022-06-27. *OpenReview.net*.

Lusch, B.; Kutz, J. N.; and Brunton, S. L. 2018. Deep learning for universal linear embeddings of nonlinear dynamics. *Nat. Commun.*, 9(1): 4950.

Mardt, A.; Pasquali, L.; Wu, H.; and Noe, F. 2018. VAMPnets for Deep Learning of Molecular Kinetics. In *Nature Communications*, volume 9, 5.

Mezić, I. 2005. Spectral Properties of Dynamical Systems, Model Reduction and Decompositions. *Nonlinear Dynamics*, 41: 309–325.

Otto, S. E.; and Rowley, C. W. 2019. Linearly Recurrent Autoencoder Networks for Learning Dynamics. *SIAM Journal on Applied Dynamical Systems*, 18(1): 558–593.

Schmid, P. J. 2010. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656: 5–28.

Tian, Y.; Chen, X.; and Ganguli, S. 2021. Understanding self-supervised learning dynamics without contrastive pairs. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 10268–10278. PMLR.

van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605.

Verdenius, S.; Zerio, A.; and Wang, R. L. M. 2024. LaT-PFN: A Joint Embedding Predictive Architecture for In-context Time-series Forecasting. arXiv:2405.10093.

Wehmeyer, C.; and Noé, F. 2018. Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *J. Chem. Phys.*, 148(24): 241703.

Yeung, E.; Kundu, S.; and Hodas, N. 2019. Learning Deep Neural Network Representations for Koopman Operators of Nonlinear Dynamical Systems. In *2019 American Control Conference (ACC)*, 4832–4839.

Yue, Z.; Wang, Y.; Duan, J.; Yang, T.; Huang, C.; Tong, Y.; and Xu, B. 2022. TS2Vec: Towards Universal Representation of Time Series. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(8): 8980–8987.