On the Internal Semantics of Time-Series Foundation Models

Anonymous Author(s)
Affiliation
Address
email

Abstract

Time-series foundation models (TSFMs) have recently emerged as a universal paradigm for learning across diverse temporal domains. Despite their empirical success, the internal mechanisms by which these models represent fundamental time-series concepts remain poorly understood. In this work, we undertake a systematic investigation of concept interpretability in TSFMs. Specifically, we examine: (i) which layers encode which concepts, (ii) whether concept parameters are linearly recoverable, (iii) how representations evolve in terms of concept disentanglement and abstraction across model depth, and (iv) how models process compositions of concepts, which serve as controlled settings for studying interaction and interference. We systematically probe these questions using layer-wise analyses, linear recoverability tests, and representation similarity measures, providing a structured account of TSFM semantics. The resulting insights show that early layers mainly capture local, time-domain patterns (e.g., AR(1), level shifts, trends), while deeper layers encode dispersion and change-time signals, with spectral and warping factors remaining the hardest to recover linearly. In compositional settings, however, probe performance degrades, revealing interference between concepts. This highlights that while atomic concepts are reliably localized, composition remains a challenge pointing to the need for composition-aware training and evaluation protocols to better align TSFMs with the structure of real-world time series.

1 Introduction

Foundation models have recently been extended to time series, where large-scale pretraining over heterogeneous temporal data yields strong zero/few-shot performance in forecasting and classification across healthcare, finance, climate, and energy [Das et al., 2023, Ansari et al., 2024, Goswami et al., 2024, Woo et al., 2024, Garza et al., 2023]. Yet, unlike language and vision, our understanding of what these models encode internally remains limited. Interpretability in NLP and CV has shown that probing methods—linear and structural probes as well as representational similarity—can localize information across layers and provide insight into model organization [Alain and Bengio, 2016, Hewitt and Manning, 2019, Kornblith et al., 2019]. For TSFMs, early studies such as [Wiliński et al., 2024] reveal block-like layer similarity and the success of latent interventions, underscoring the value of probing. Complementary instance-level explanations in time series, e.g., saliency, attribution, and shapelets, offer rationales for individual predictions but do not illuminate model-wide semantics [Ismail et al., 2020, Grabocka et al., 2014].

This gap makes it crucial to investigate how TSFMs represent fundamental time-series phenomena. We address this by studying *concept interpretability* in TSFMs across seven canonical concepts that span stochastic, structural, and spectral behavior: *AR1*, *Level Shift*, *Random Walk*, *Spectral*, *Time*

Under review at the NeurIPS 2025 Workshop on Recent Advances in Time Series Foundation Models (BERT^{2}S). Do not distribute.

Warped, *Trend*, and *Variance Shift*. Our analysis is guided by four central questions: **RQ1**—where concepts localize across layers; **RQ2**—whether concept parameters are linearly recoverable from intermediate embeddings; **RQ3**—how disentanglement and abstraction evolve with depth; and **RQ4**—how compositions of concepts interact.

Methodologically, we adapt established tools—linear probes, structural probes, and CKA—while explicitly tailoring them to measure concept presence and parameter recoverability [Alain and Bengio, 2016, Hewitt and Manning, 2019, Kornblith et al., 2019]. Although these diagnostics are widely used in other domains, their systematic application to a controlled, diverse suite of time-series concepts provides fresh evidence about the inductive biases and limitations of representative TSFMs [Das et al., 2023, Ansari et al., 2024, Goswami et al., 2024, Woo et al., 2024, Garza et al., 2023, Wiliński et al., 2024].

Contributions. (i) A concept-centric probing benchmark for TSFMs spanning seven canonical concepts; (ii) diagnostic tasks targeting concept localization, parameter recoverability, and compositional interaction; (iii) empirical findings that uncover inductive biases and failure modes, informing architectural choices, training curricula, and evaluation practice.

2 Methods

Layer-wise Concept Probing. *RQ1* and *RQ2* ask which layers encode which concepts and whether their parameters are linearly recoverable. To answer this, we apply linear probes across layers, following a methodology widely used in language models to reveal whether syntax or semantics emerges at specific depths. For time series, this allows us to pinpoint where autoregressive structure, spectral frequency, or trend parameters become accessible. Given a synthetic dataset $\mathbf{X} \in \mathbb{R}^{S \times V}$ with generative parameters θ, a TSFM with L layers produces hidden states $\mathbf{H}^{(l)} = f^{(l)}(\mathbf{X}) \in \mathbb{R}^{S \times d}$, pooled into $\mathbf{z}^{(l)} = \text{Pool}(\mathbf{H}^{(l)}) \in \mathbb{R}^d$. A linear probe then predicts parameters via $\hat{\theta}^{(l)} = \mathbf{W}^{(l)}\mathbf{z}^{(l)} + \mathbf{b}^{(l)}$. Performance is measured by mean squared error, $\mathcal{L}^{(l)} = \frac{1}{N}\sum_{i=1}^{N} \|\theta_i - \hat{\theta}_i^{(l)}\|^2$, quantifying parameter recoverability across depth.

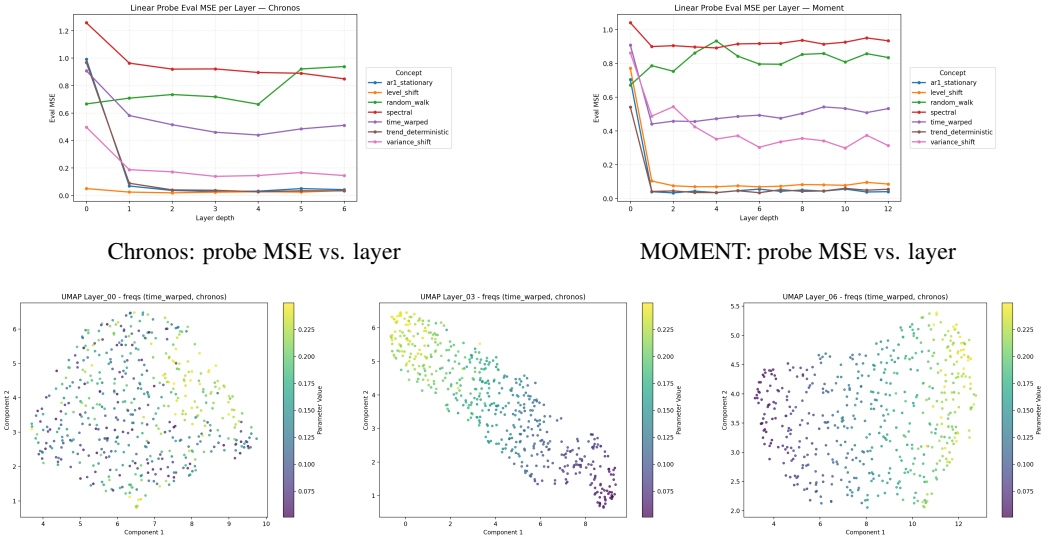
Concept Representation. *RQ3* concerns how representations evolve across depth—whether they become more abstract or more disentangled. In computer vision, representational similarity analyses reveal shifts from low-level edges to object-level semantics. We adopt a similar lens for TSFMs, examining whether different time-series concepts occupy distinct or overlapping regions in embedding space, and how this structure changes across layers. To examine representational similarity across concepts and layers, we compute centered kernel alignment (CKA) between embedding sets $\mathbf{H}^{(l_1)}$ and $\mathbf{H}^{(l_2)}$: $\text{CKA}(\mathbf{H}^{(l_1)}, \mathbf{H}^{(l_2)}) = \frac{\|\mathbf{H}^{(l_1)\top}\mathbf{H}^{(l_2)}\|_F^2}{\|\mathbf{H}^{(l_1)\top}\mathbf{H}^{(l_1)}\|_F \, \|\mathbf{H}^{(l_2)\top}\mathbf{H}^{(l_2)}\|_F}$. Additionally, we visualize embeddings via PCA, UMAP, and t-SNE [Jolliffe, 2002, McInnes et al., 2018, van der Maaten and Hinton, 2008] applied to pooled vectors $\mathbf{z}^{(l)}$, allowing inspection of cluster structure and concept separation.

Concept Composition. RQ4 asks how TSFMs handle compositions and whether concept-specific information transfers to mixtures. We adopt a two-step *probe-transfer* protocol: (i) train layer-wise linear probes on *atomic* data for each concept C_j to predict its parameters θ_j (backbone frozen); (ii) evaluate these frozen probes for C_1 and C_2 on *composite* series to assess whether the original parameters remain linearly recoverable. We report per-layer MSE on composites.

We study two families of compositions: *structured* (segment-wise interleaving with continuity preservation) and *functional* (additive mixing, optionally with per-series normalization and mixing coefficients α). Full construction details, masks, and sampling ranges are provided in Appendix D.

3 Results and Discussions

RQ1 & RQ2: Which layers encode which concepts, and are parameters linearly recoverable?
UMAP–probe alignment. We analyze UMAP embeddings of latents corresponding to each concept across layers. Compact, well-ordered UMAP structures correlate with lower linear probe MSEs for the associated concept parameter, suggesting that when a model internally learns a concept, its latent representations become localized. Moreover, when the probed parameter aligns with a smooth gradient along the UMAP manifold, probe errors are further reduced—indicating that the model has not only captured the concept but also learned to parameterize it with respect to that



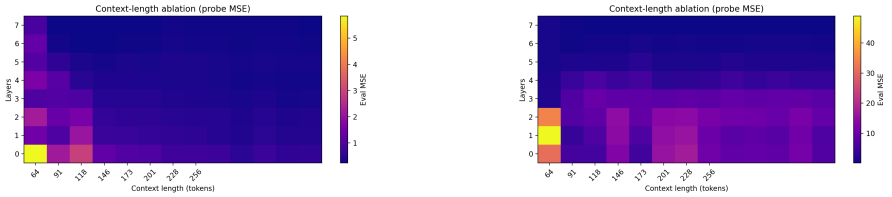
Chronos: probe MSE vs. layer

MOMENT: probe MSE vs. layer

Figure 1: Layer-wise concept probing summary. Top: linear-probe MSE across layers for Chronos (left) and MOMENT (right); each curve corresponds to one concept (legend shared). Bottom: UMAP of pooled embeddings at early, mid, and late layers, time-warp concept. Best viewed in color.

control variable. Such alignment could be particularly useful for applications that steer activations conditionally. In practice, we observe lower probe MSEs for structural and time-domain concepts such as AR(1) coefficient, trend slope, and level shifts. By contrast, spectral and time-warping concepts yield fragmented or tangled UMAP structures and higher probe errors, consistent with non-linear entanglement that resists linear recovery (see Fig.1 and AppendixG).

Model comparison and depth. Under identical setups, CHRONOS exhibits more linearly recoverable and better organized (UMAP) representations than MOMENT across all evaluated concepts. Most tasks peak early—typically around the second transformer layer—after which performance plateaus. In contrast, dispersion and change-point phenomena (e.g., variance shift) improve monotonically with depth and are localized in later layers (Fig. 1).



(a) Time Warped Concept

(b) Variance Shift Concept

Figure 2: Context Length ablations on MOMENT

RQ3: How do representations evolve with depth? From the UMAP snapshots (see Figures in the Appendix G, for e.g. Figure 12 and Figure 13) we can see increasing cluster separation from early to late layers, *which indicates that Early layers reflect locally volatile structure; mid layers show partial disentanglement; late layers consolidate concept-level separation while compressing intra-concept variance*. This also aligns with the decreasing MSE after layer 1, indicating a shift from generic to concept-aligned features.

We further assess each layer's reliance on temporal context by cropping inputs to multiple fractions (25-100%), extracting pooled embeddings, and evaluating the pre-trained per-layer linear probes against the concept targets. From Figure 48a and Figure 48b we can see how MSE changes with available history; *deeper layers improve as context grows (encoding longer-range dynamics) compared to relatively less improvement in early layers (capturing short, local structure).*

RQ4: How are concept compositions represented? While TSFMs can localize and represent atomic time-series concepts, real-world data often consists of compositions of multiple such concepts. To study the behavior of TSFMs under composite concepts, we consider the following experiments: **(a) Vector Arithmetic** — Inspired by word embedding compositionality, we test whether TSFM embeddings exhibit similar additive properties. Specifically, we evaluate whether the element-wise sum of atomic concept embeddings ($\mathbf{emb_1} + \mathbf{emb_2}$) approximates the embedding of their composite concept ($\mathbf{emb_3}$) using cosine similarity and relative distance metrics across model layers. **(b) Temporal Alignment Analysis** — Since time-series have inherent temporal structure, we test compositional stability across different sequence lengths (32, 64, 128, 256 timesteps). This evaluates whether compositional relationships hold consistently across temporal horizons or are sensitive to sequence length.

Figure 3 **reveals** strong compositional properties in TSFMs, with cosine similarities approaching 1.0 across most layers, indicating that atomic concept embeddings combine nearly linearly ($\mathbf{emb_1} + \mathbf{emb_2} \approx \mathbf{emb_3}$). Performance degradation in initial and final layers suggests that early representations lack full compositional structure, while deeper layers specialize in task-specific features that deviate from additive composition. The anomalous behavior of *spectral+level_shift*, which shows substantially higher relative distances, indicates non-linear interactions between concepts with fundamentally different temporal characteristics—frequency-domain properties versus abrupt discontinuities. Overall, TSFMs learn compositional representations similar to word embeddings for most concept pairs, with notable exceptions requiring more sophisticated composition mechanisms for temporally disparate concepts.

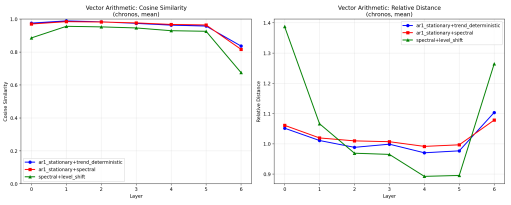


Figure 3: Vector arithmetic experiments with CHRONOS. Atomic embeddings combine nearly linearly ($\mathbf{emb_1} + \mathbf{emb_2} \approx \mathbf{emb_3}$), except for temporally disparate concept pairs.

The temporal alignment analysis results (see Figure 4) **demonstrate** robust compositional stability across sequence lengths, with consistently high similarities throughout most layers and temporal horizons. Reduced similarity at shorter sequences (32–128) in the initial and final layers suggests that compositional understanding requires sufficient temporal context to emerge and stabilize. The uniformly high performance at longer sequences confirms that TSFMs' compositional properties are temporally robust once adequate context is provided. Please refer to Appendix H for add. results.

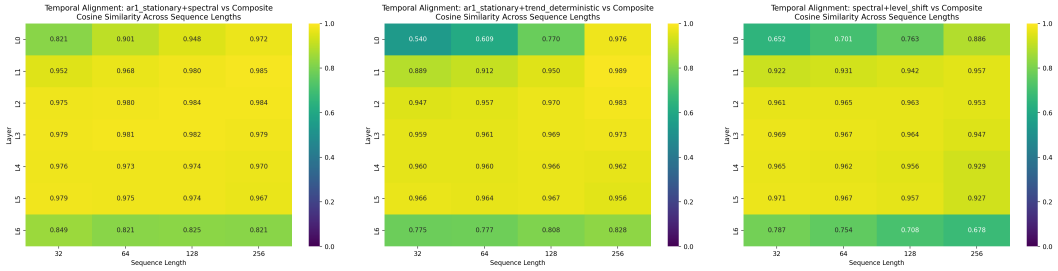


Figure 4: Chronos – Temporal alignment experiments. We show stability of compositional relationships across multiple atomic-concept pairs.

4 Conclusion and Future Work

We presented a probe-based analysis of time-series foundation models across seven canonical concepts. Early layers expose local, time domain structure (AR(1), level shift, trend), deeper layers localize dispersion and change-time signals, and spectral/warping factors are the least linearly accessible. On compositions, TSFMs exhibit strong linear compositional properties across most layers and concept pairs. Future works could extend to multivariate, irregular, and real datasets; adopt controlled-capacity non-linear or causal probes; design objectives and architectures that better linearize phase and time-warping, and work on non-linear conditional steering.

References

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016. doi: 10.48550/arXiv.1610.01644. URL https://arxiv.org/abs/1610.01644.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024. doi: 10.48550/arXiv.2403.07815. URL https://arxiv.org/abs/2403.07815.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2023. doi: 10.48550/arXiv.2310.10688. URL https://arxiv.org/abs/2310.10688.

Azul Garza, Cristian Challu, and Max Mergenthaler-Canseco. Timegpt-1. *arXiv preprint arXiv:2310.03589*, 2023. doi: 10.48550/arXiv.2310.03589. URL https://arxiv.org/abs/2310.03589.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 16115–16152. PMLR, 2024. URL https://proceedings.mlr.press/v235/goswami24a.html.

Josif Grabocka, Nicolas Schilling, Martin Wistuba, and Lars Schmidt-Thieme. Learning time-series shapelets. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 392–401. ACM, 2014. doi: 10.1145/2623330.2623613. URL https://dl.acm.org/doi/10.1145/2623330.2623613.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4129–4138, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1419. URL https://aclanthology.org/N19-1419/.

Aya Abdelsalam Ismail, Mohamed Gunady, Héctor Corrada Bravo, and Soheil Feizi. Benchmarking deep learning interpretability in time series predictions. In *Advances in Neural Information Processing Systems*, volume 33, pages 6441–6452, 2020. URL https://proceedings.neurips.cc/paper/2020/file/47a3893cc405396a5c30d91320572d6d-Paper.pdf.

Ian T. Jolliffe. *Principal Component Analysis*. Springer, New York, NY, second edition, 2002. doi: 10.1007/b98835.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3519–3529. PMLR, 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. URL https://arxiv.org/abs/1802.03426.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008. URL https://www.jmlr.org/papers/v9/vandermaaten08a.html.

Michał Wiliński, Mononito Goswami, Nina Żukowska, Willa Potosnak, and Artur Dubrawski. Exploring representations and interventions in time series foundation models. *arXiv preprint arXiv:2409.12915*, 2024. URL https://arxiv.org/abs/2409.12915.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 53140–53164. PMLR, 2024. URL https://proceedings.mlr.press/v235/woo24a.html.

A Related Works

Time-series foundation models. Recent TSFMs demonstrate strong zero/few-shot performance via large-scale pretraining and task-agnostic architectures. Representative families include TimesFM (decoder-only with patched attention), Chronos (tokenized values with T5-style training), MOMENT (open models and the Time Series Pile), Moirai (masked-encoder universal forecaster), and TimeGPT (closed-source API). These works establish the empirical promise of TSFMs but do not characterize concept-level internal semantics. [Das et al., 2023, Ansari et al., 2024, Goswami et al., 2024, Woo et al., 2024, Garza et al., 2023]

Probing and representational similarity. Linear probes and related diagnostic tools are widely used to localize information across layers in deep networks, originating with linear classifier probes and extended by structural probes in NLP to test linear recoverability of syntax. Centered Kernel Alignment (CKA) is commonly used to compare layer representations within and across models due to its invariances and robustness relative to earlier CCA-style measures. Our study adapts these established tools to TSFMs and focuses them on time-series concepts and parameters. [Alain and Bengio, 2016, Hewitt and Manning, 2019, Kornblith et al., 2019]

Interpreting TSFMs and time-series models. Closest to our work, Wiliński et al. analyze internal redundancy and concept steering in TSFMs, reporting block-like layer similarity and latent-space interventions; we complement this by centering *concept parameters*, layer-wise recoverability, and controlled compositions. Broader interpretability for time series has emphasized saliency/attribution and shapelet-based explanations; these provide instance-level rationales, whereas our focus is on *representation-level* concept encoding across depth. [Wiliński et al., 2024, Ismail et al., 2020, Grabocka et al., 2014]

B Experimental Setup

Datasets. We evaluate seven synthetic concepts: AR(1), Level Shift, Random Walk, Spectral (sum of sinusoids), Time-Warped Sinusoid, Deterministic Trend, and Variance Shift. Generation procedures, parameter ranges, and normalization rules follow Appendix D (Dataset Generation and Description). We additionally construct compositional datasets by pairing two base concepts.

Models. We use publicly released checkpoints of two time-series foundation models: Chronos and MOMENT since both are T-5 like models transformer architecture. Model weights are frozen and no finetuning is performed.

Evaluation and reporting. We use an 80/20 train/validation split for each concept and composition. Metric is mean squared error (MSE) for parameter recovery.

C Dimensionality Reduction Techniques

Principal Component Analysis (PCA). Given pooled representations $\{\mathbf{z}_i^{(l)}\}_{i=1}^N$, we compute the empirical covariance matrix

$$\mathbf{\Sigma}^{(l)} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{z}_i^{(l)} - \bar{\mathbf{z}}^{(l)}\right)\left(\mathbf{z}_i^{(l)} - \bar{\mathbf{z}}^{(l)}\right)^\top.$$

Eigen-decomposition yields orthogonal axes capturing the largest variance directions:

$$\mathbf{\Sigma}^{(l)}\mathbf{u}_k = \lambda_k \mathbf{u}_k, \quad \lambda_1 \geq \lambda_2 \geq \dots.$$

These principal axes reveal which parameters dominate the representation space and whether layers compress or expand information.

t-SNE. To assess local neighborhoods, we apply t-distributed Stochastic Neighbor Embedding (t-SNE), which constructs pairwise similarities in high- and low-dimensional spaces. For two points $\mathbf{z}_i, \mathbf{z}_j$, their similarity in the original space is

$$p_{ij} = \frac{\exp(-\|\mathbf{z}_i - \mathbf{z}_j\|^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-\|\mathbf{z}_i - \mathbf{z}_k\|^2/2\sigma_i^2)},$$

while in 2D space the similarity is

$$q_{ij} = \frac{\left(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2\right)^{-1}}.$$

t-SNE minimizes the Kullback–Leibler divergence:

$$\mathrm{KL}(P\|Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}.$$

This highlights fine-grained clusters and separability of parameter values.

UMAP. Uniform Manifold Approximation and Projection seeks a balance between local and global structure. It constructs a weighted k-nearest-neighbor graph and optimizes a low-dimensional embedding $\{\mathbf{y}_i\}$ by minimizing the cross-entropy between high- and low-dimensional fuzzy simplicial sets:

$$\mathcal{L}_{\text{UMAP}} = \sum_{(i,j)} w_{ij} \log \sigma(\|\mathbf{y}_i - \mathbf{y}_j\|) + (1 - w_{ij}) \log(1 - \sigma(\|\mathbf{y}_i - \mathbf{y}_j\|)),$$

where σ is a differentiable approximation of a step function. UMAP can reveal concept families and hierarchical relationships (e.g., stationary vs. nonstationary).

These projections provide intuition about the embedding geometry—global variance (PCA), local clusters (t-SNE), and local-global trade-offs (UMAP)—which the linear probes then quantify.

D Synthetic Datasets

This section summarizes the synthetic time–series concepts used in our experiments, their generating equations, and key parameters. Unless noted, ε_t denotes i.i.d. Gaussian noise.

D.1 AR(1) (Stationary)

$$x_t = \phi\, x_{t-1} + \varepsilon_t, \quad |\phi| < 1, \tag{1}$$

$$\varepsilon_t \sim \mathcal{N}(0, \sigma^2), \quad x_0 \text{ drawn from the stationary distribution.} \tag{2}$$

Parameters: autoregressive coefficient ϕ (sampled from an interval), innovation std σ. Default normalization: per-series z-score.

D.2 Level Shift

$$x_t = \eta_t + \Delta\, \mathbf{1}\{t \geq \tau\}, \quad \eta_t \sim \mathcal{N}(0, \text{noise\_std}^2). \tag{3}$$

Parameters: signed shift magnitude Δ, changepoint τ, noise std. Default normalization: none (scale encodes the signal).

D.3 Random Walk (With Drift)

$$x_t = x_{t-1} + \mu + \varepsilon_t, \tag{4}$$

$$\varepsilon_t \sim \mathcal{N}(0, \sigma^2). \tag{5}$$

Parameters: drift μ, innovation std. Default normalization: none.

D.4 Spectral (Sum of Sinusoids)

$$x_t = \sum_{j=1}^{k} a_j \sin\left(2\pi f_j t + \phi_j\right) + \varepsilon_t, \quad 0 < f_j < 0.5. \tag{6}$$

Parameters: number of components $k \in \{1, \ldots, k_{\max}\}$; amplitudes a_j; frequencies f_j sampled from $[\text{freq\_low}, \text{freq\_high}]$; phases $\phi_j \sim \text{Uniform}(0, 2\pi)$; noise std. Default normalization: per-series z-score.

D.5 Time-Warped Sinusoid

Generate a base sinusoid $b_t = \sin(2\pi f t + \phi)$, draw positive steps from a Gamma distribution, form a monotone cumulative warp u rescaled to $[0, T-1]$, then reinterpolate back to the regular grid:

$$x_t = \text{interp}(t, u, b) + \varepsilon_t. \tag{7}$$

Parameters: base frequency f, phase ϕ, warp strength, noise std. Default normalization: per-series z-score.

D.6 Deterministic Trend

$$x_t = \beta\, t + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \text{noise\_std}^2). \tag{8}$$

Parameters: slope β, noise std. Default normalization: per-series z-score.

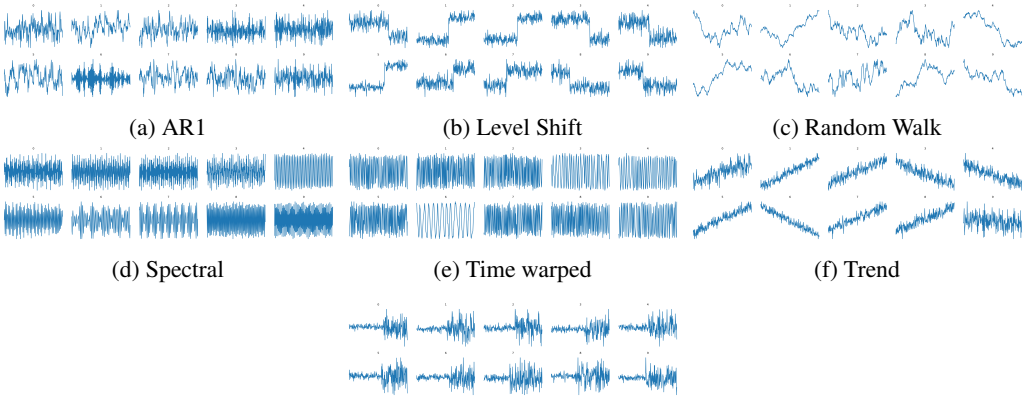
D.7 Variance Shift

$$x_t \sim \begin{cases} \mathcal{N}(0, \sigma_1^2), & t < \tau, \\ \mathcal{N}(0, \sigma_2^2), & t \geq \tau. \end{cases} \tag{9}$$

Parameters: changepoint τ, standard deviations σ_1, σ_2. Default normalization: none.

Notes on Normalization Concepts where magnitude/level is the signal (e.g., level or variance shift, random walk) use no normalization by default; others use per-series z-scoring. See the code reference (`concepts_dataset.py`) for full details and sampling ranges.

D.8 Time-series Concepts



(a) AR1 (b) Level Shift (c) Random Walk

(d) Spectral (e) Time warped (f) Trend

(g) Variance Shift

Figure 5: Visualization of the synthetic time-series samples generated

D.9 Time series composition

Let $\mathcal{T}1 = \{T_1^{(i)}\}i = 1^N$ and $\mathcal{T}2 = \{T_2^{(i)}\}i = 1^N$ be two sets of time series generated from concepts C_1 and C_2 respectively, where each $T_j^{(i)} \in \mathbb{R}^T$.

Structured Composition. Temporal interleaving with continuity constraints, preserving local concept characteristics in different time segments.

For each sample i, we generate breakpoints a_i, b_i where:

$$a_i \sim \mathcal{U}(\lfloor \alpha_{\text{low}} \cdot T \rfloor, \lfloor \alpha_{\text{high}} \cdot T \rfloor)$$

$$b_i \sim \mathcal{U}(\lfloor \beta_{\text{low}} \cdot T \rfloor, \lfloor \beta_{\text{high}} \cdot T \rfloor)$$

with constraints $0 \leq a_i < b_i \leq T$ and default ranges $\alpha_{\text{low}} = 0.2, \alpha_{\text{high}} = 0.4, \beta_{\text{low}} = 0.6, \beta_{\text{high}} = 0.8$. The structured compositional series $X_{\text{struct}}^{(i)}$ is defined as:

$$X_{\text{struct}}^{(i)}[t] = \begin{cases} T_1^{(i)}[t] & \text{if } t < a_i \\ T_2^{(i)}[t] + \delta_1^{(i)} & \text{if } a_i \leq t < b_i \\ T_1^{(i)}[t] + \delta_2^{(i)} & \text{if } t \geq b_i \end{cases}$$

where the continuity offsets are:

$$\delta_1^{(i)} = T_1^{(i)}[a_i] - T_2^{(i)}[a_i]$$

$$\delta_2^{(i)} = T_2^{(i)}[b_i] - T_1^{(i)}[b_i] + \delta_1^{(i)}$$

The corresponding mask $M^{(i)} \in \{0, 1\}^T$ indicates the source concept:

$$M^{(i)}[t] = \begin{cases} 0 & \text{if } t < a_i \text{ or } t > b_i \text{ (from } C_1\text{)} \\ 1 & \text{if } a_i \leq t \leq b_i \text{ (from } C_2\text{)} \end{cases}$$

Functional Composition. Elementwise addition creating global interaction between concepts throughout the entire time series Both approaches generate datasets containing the composed series X, original component series T_1, T_2, and metadata preserving the generative parameters from both source concepts.

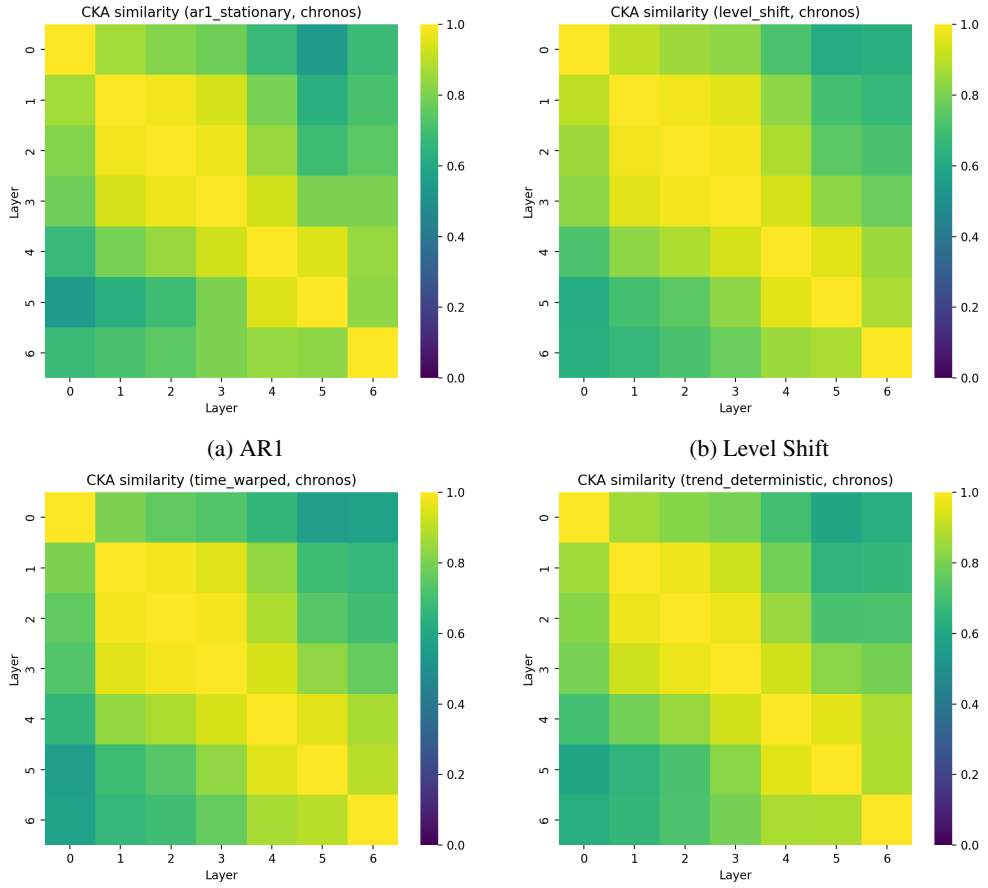
For functional composition, we first optionally normalize each time series:

$$\tilde{T}j^{(i)} = \begin{cases} \frac{T_j^{(i)} - \mu_j^{(i)}}{\sigma_j^{(i)}} & \text{if normalize= True} \\ T_j^{(i)} & \text{otherwise} \end{cases}$$

where $\mu_j^{(i)} = \frac{1}{T}\sum_{t=1}^{T} T_j^{(i)}[t]$ and $\sigma_j^{(i)} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(T_j^{(i)}[t] - \mu_j^{(i)})^2}$. The functional compositional series is then:

$$X_{\text{func}}^{(i)} = \tilde{T}1^{(i)} + \tilde{T}_2^{(i)}$$

E Layer Representation Similarity



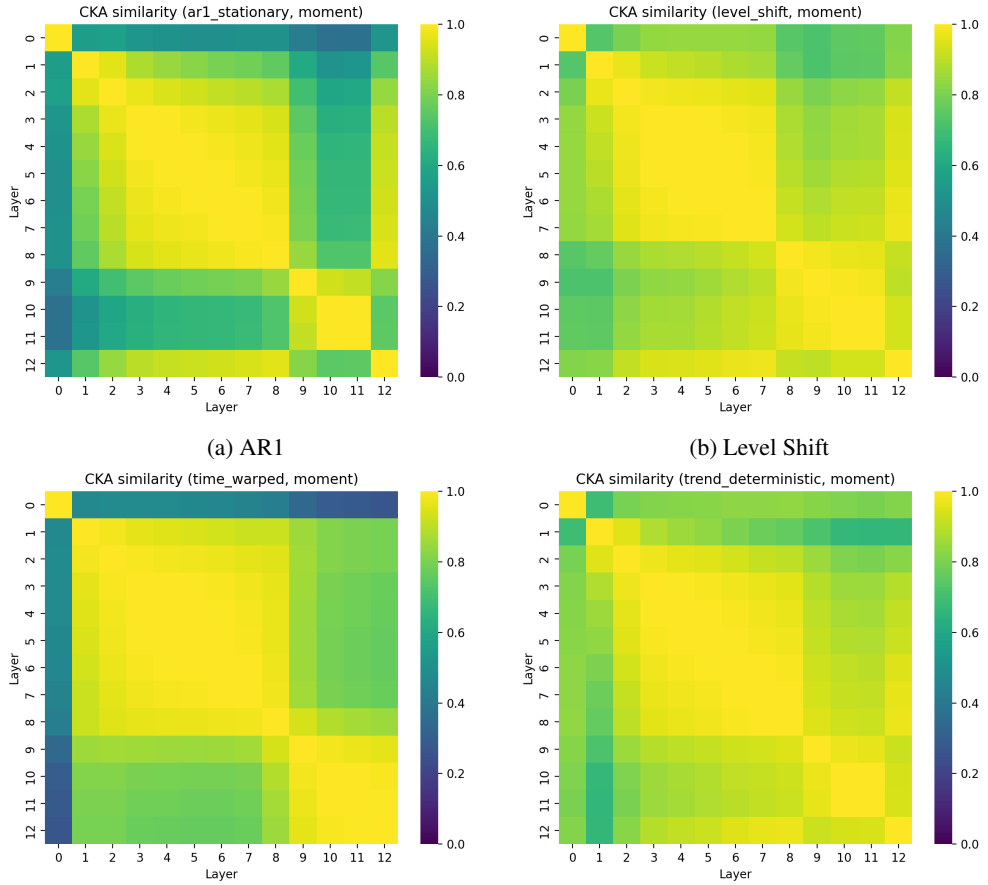
(a) AR1

(b) Level Shift

(c) Time warped

(d) Trend

Figure 6: CKA Similarity among layers of Chronos TSFM



(a) AR1

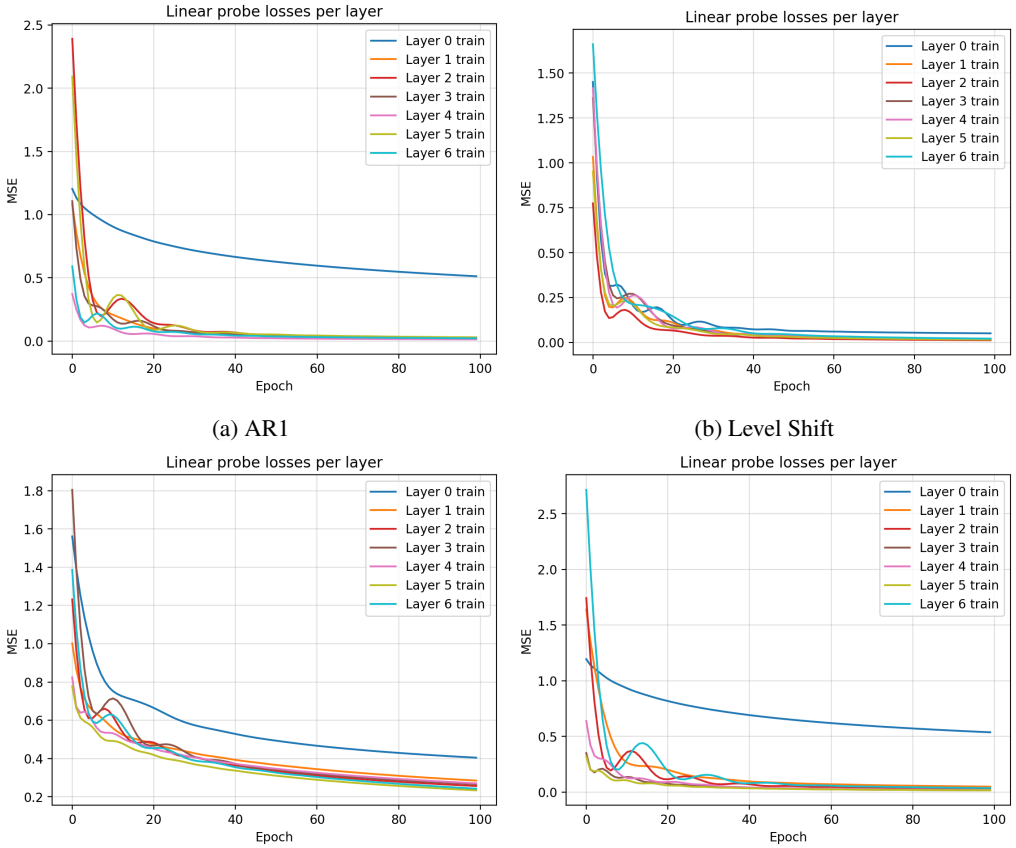
(b) Level Shift

(c) Time warped

(d) Trend

Figure 7: CKA Similarity among layers of MOMENT TSFM

F Linear Probe Loss



(a) AR1

(b) Level Shift

(c) Time warped

(d) Trend

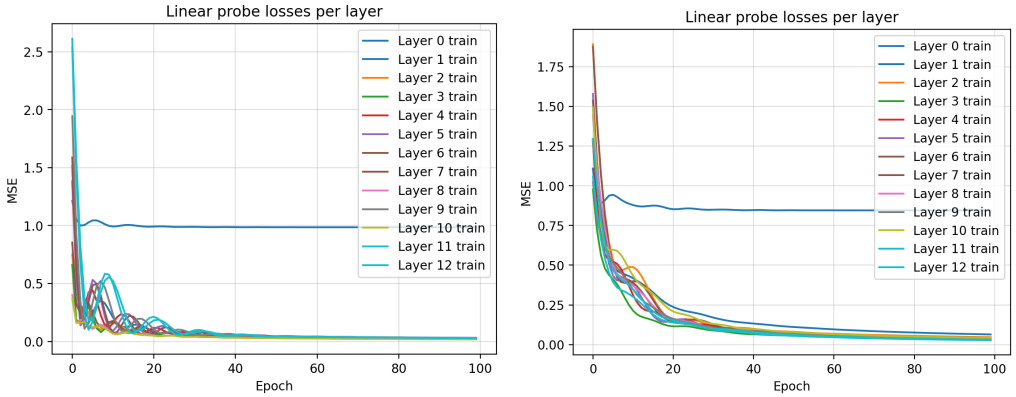
Figure 8: Layer-wise Loss in Chronos TSFM

G Layerwise Respresentation Visualization

This section summarizes layerwise embeddings visualized via PCA, t-SNE, and UMAP for each concept and model. We show triplets of layers per method.

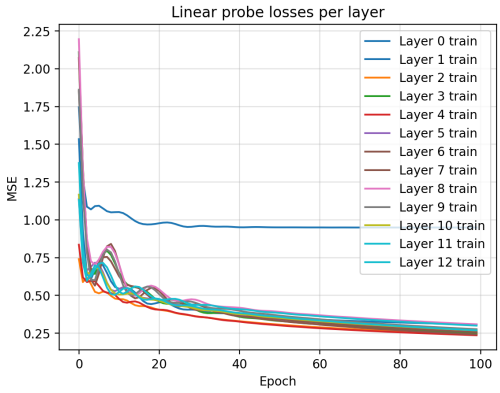
G.1 AR(1) (Stationary)

Moment (parameter: ϕ).

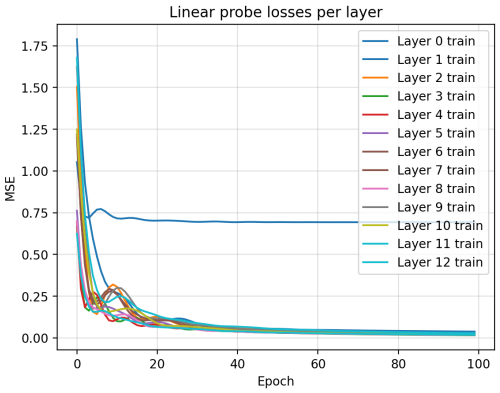


(a) AR1

(b) Level Shift



(c) Time warped



(d) Trend

Figure 9: Layer-wise Loss in MOMENT TSFM

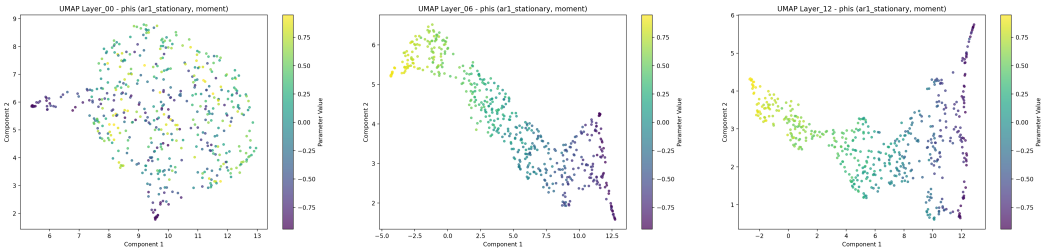


Figure 10: AR(1) — Moment — UMAP (Layers 00/06/12)

Chronos (parameter: ϕ).

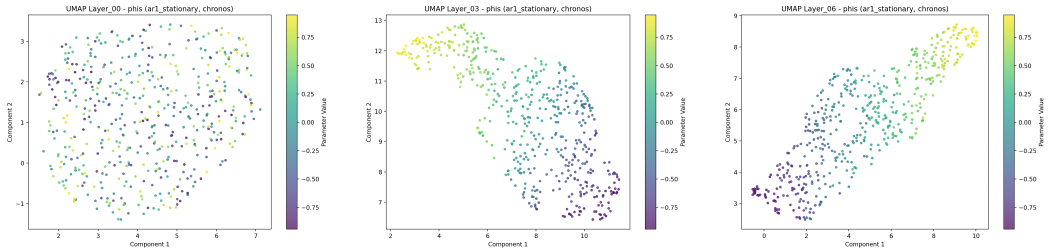


Figure 11: AR(1) — Chronos — UMAP (Layers 00/03/06)

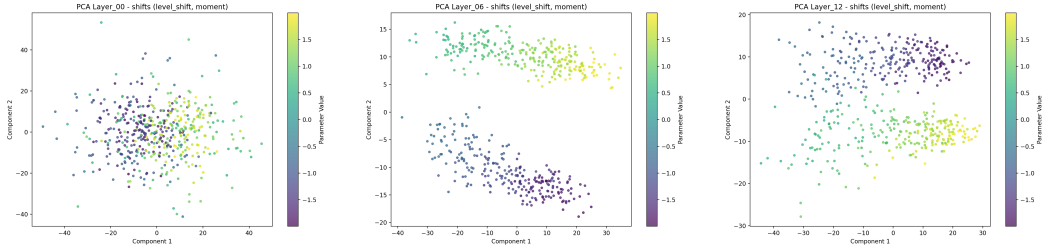


Figure 12: Level Shift — Moment — Shift — PCA (Layers 00/06/12)

G.2 Level Shift

Moment (parameters: shift, τ).

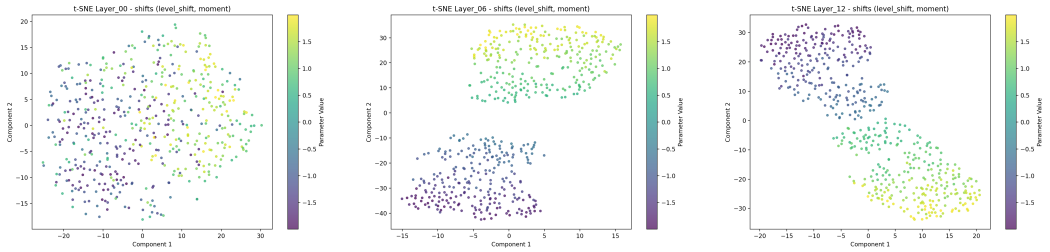


Figure 13: Level Shift — Moment — Shift — t-SNE (Layers 00/06/12)

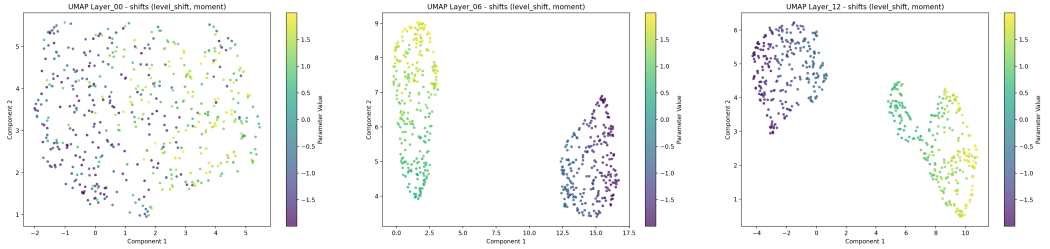


Figure 14: Level Shift — Moment — Shift — UMAP (Layers 00/06/12)

Chronos (parameters: shift, τ).

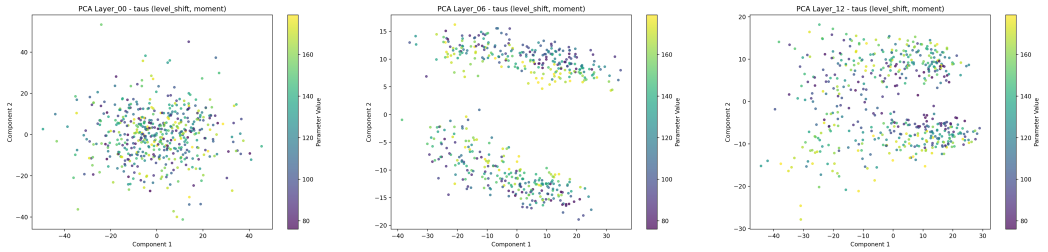


Figure 15: Level Shift — Moment — τ — PCA (Layers 00/06/12)

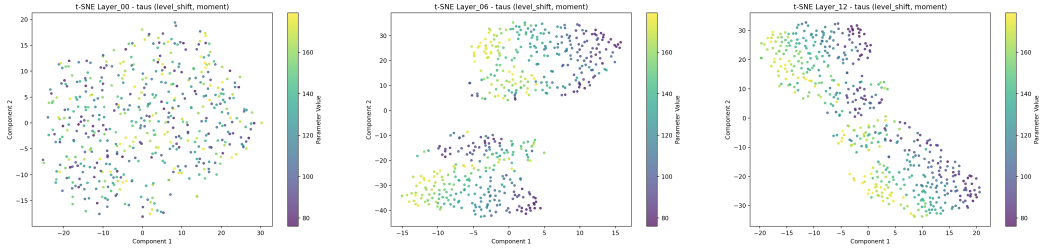


Figure 16: Level Shift — Moment — τ — t-SNE (Layers 00/06/12)

G.3 Random Walk

Chronos (parameter: drift).

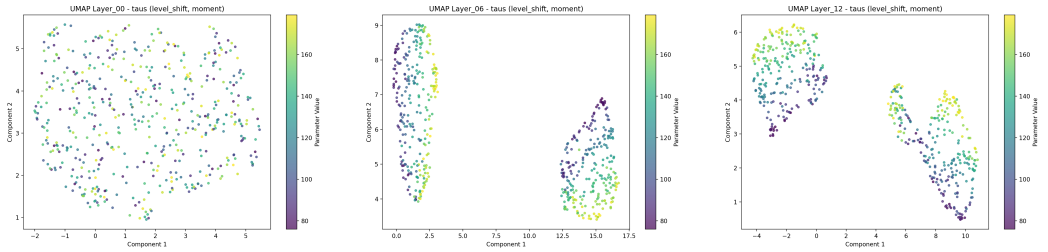


Figure 17: Level Shift — Moment — τ — UMAP (Layers 00/06/12)

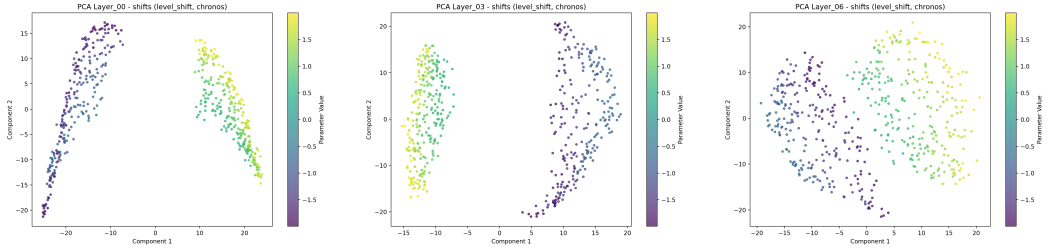


Figure 18: Level Shift — Chronos — Shift — PCA (Layers 00/03/06)

Moment (parameter: drift).

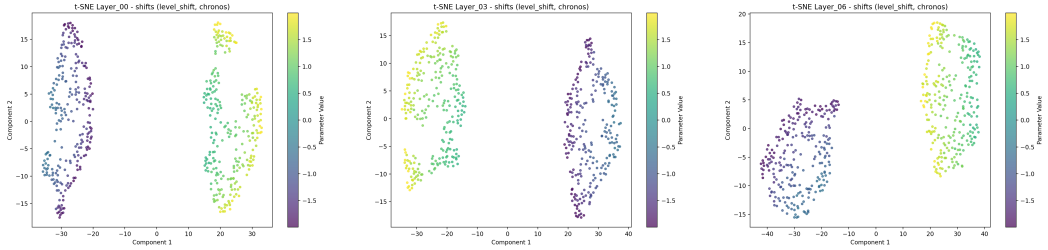


Figure 19: Level Shift — Chronos — Shift — t-SNE (Layers 00/03/06)

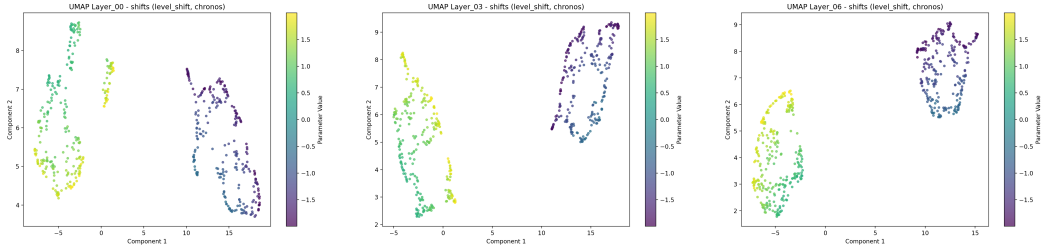


Figure 20: Level Shift — Chronos — Shift — UMAP (Layers 00/03/06)

G.4 Spectral (Sum of Sinusoids)

Chronos (frequency).

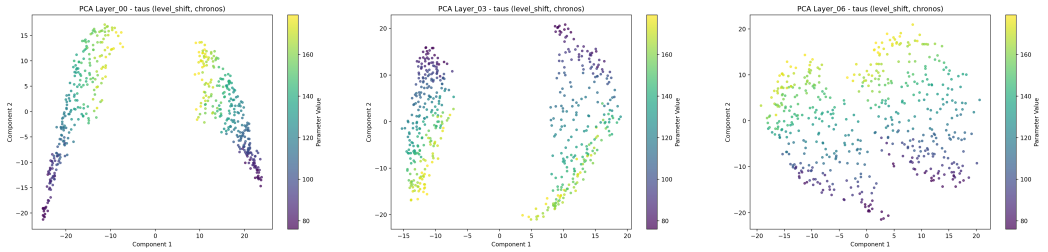


Figure 21: Level Shift — Chronos — τ — PCA (Layers 00/03/06)

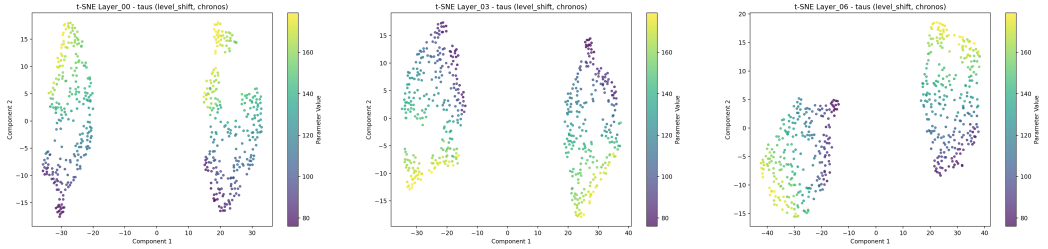


Figure 22: Level Shift — Chronos — τ — t-SNE (Layers 00/03/06)

Moment (frequency).

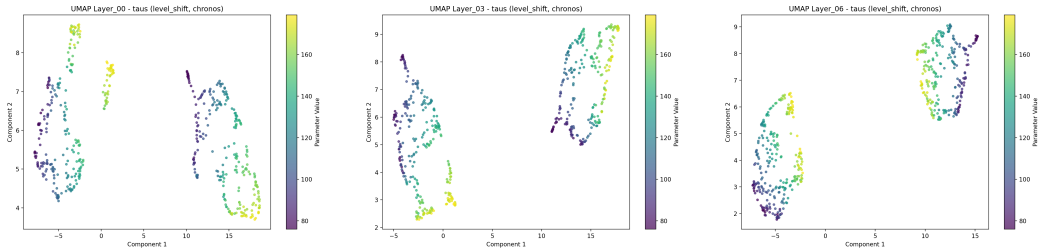


Figure 23: Level Shift — Chronos — τ — UMAP (Layers 00/03/06)

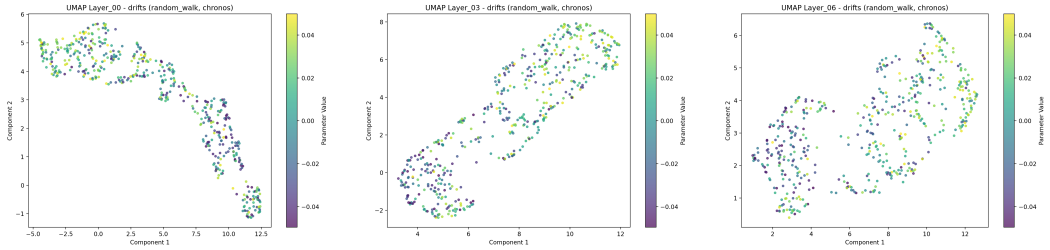


Figure 24: Random Walk — Chronos — UMAP (Layers 00/03/06)

G.5 Time-Warped Sinusoid

Moment (freq).

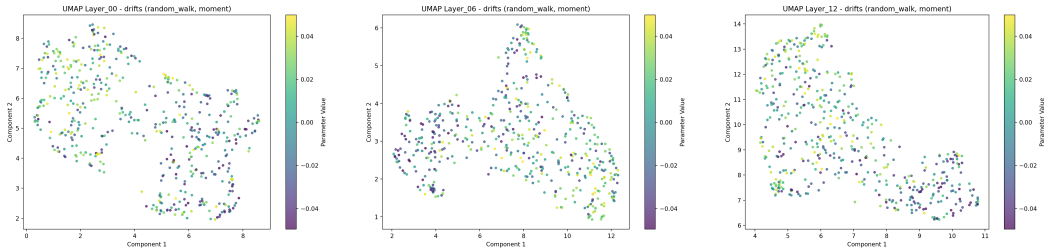


Figure 25: Random Walk — Moment — UMAP (Layers 00/06/12)

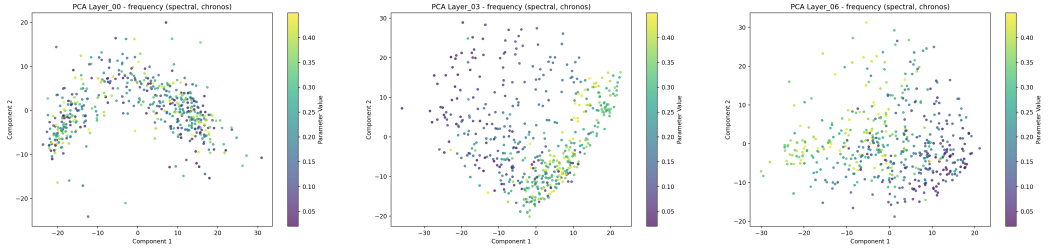


Figure 26: Spectral — Chronos — Frequency — PCA (Layers 00/03/06)

Chronos (freq).

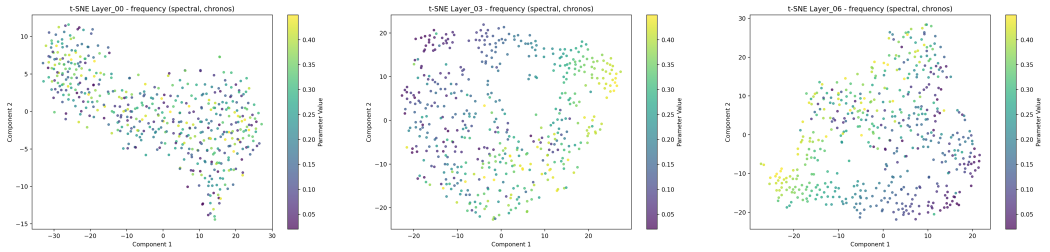


Figure 27: Spectral — Chronos — Frequency — t-SNE (Layers 00/03/06)

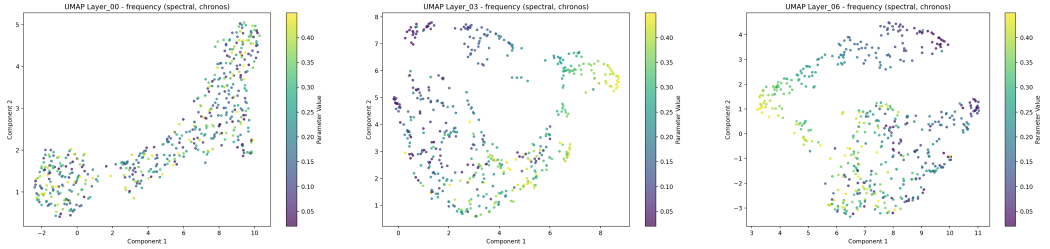


Figure 28: Spectral — Chronos — Frequency — UMAP (Layers 00/03/06)

G.6 Deterministic Trend

Moment (slope).

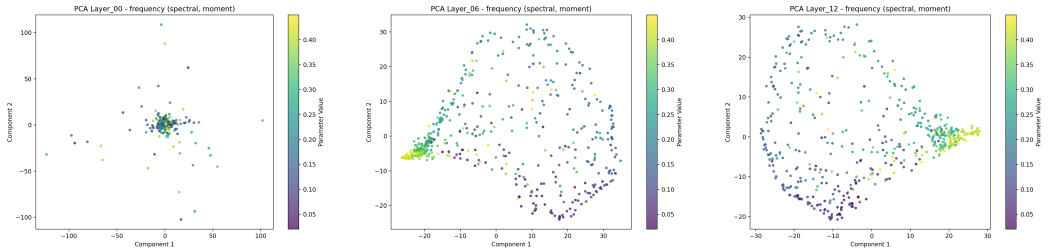


Figure 29: Spectral — Moment — Frequency — PCA (Layers 00/06/12)

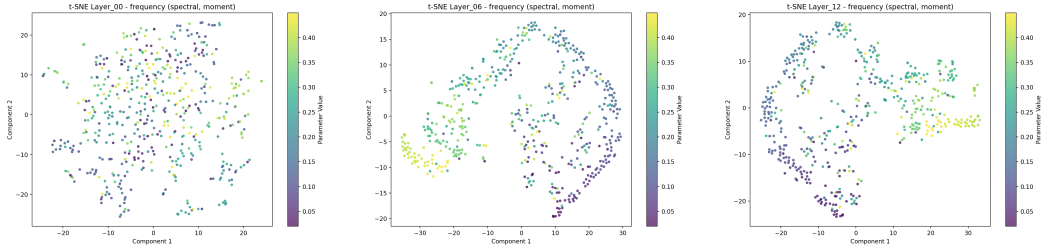


Figure 30: Spectral — Moment — Frequency — t-SNE (Layers 00/06/12)

Chronos (slope).

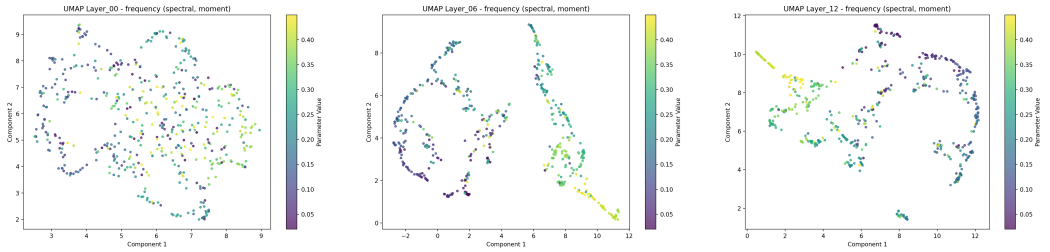


Figure 31: Spectral — Moment — Frequency — UMAP (Layers 00/06/12)

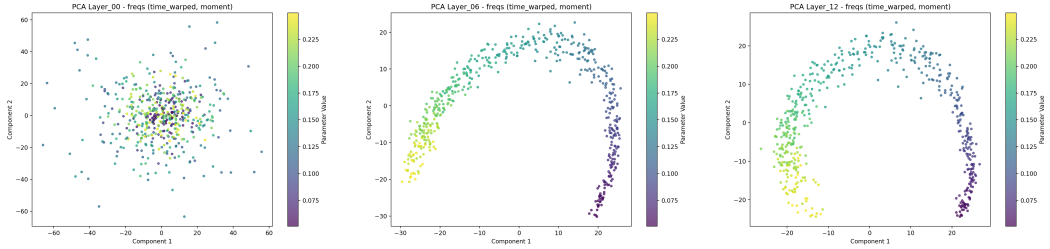


Figure 32: Time-Warped — Moment — Freqs — PCA (Layers 00/06/12)

G.7 Variance Shift

Chronos (τ).

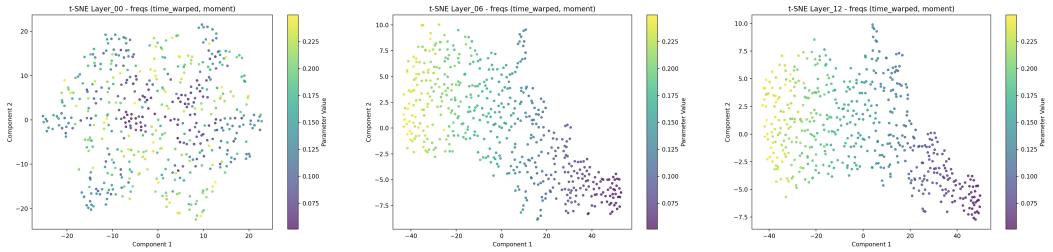


Figure 33: Time-Warped — Moment — Freqs — t-SNE (Layers 00/06/12)

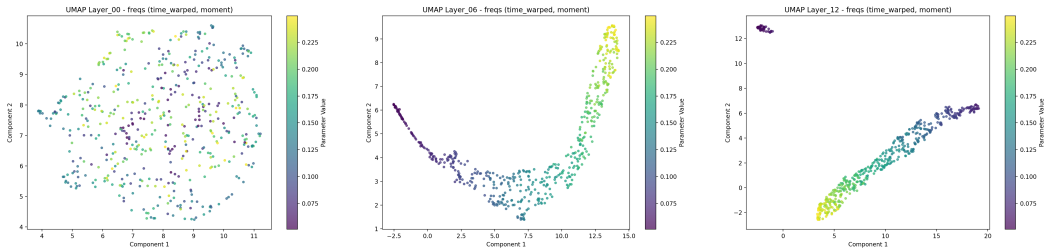


Figure 34: Time-Warped — Moment — Freqs — UMAP (Layers 00/06/12)

Moment (τ).

H Compositionality results

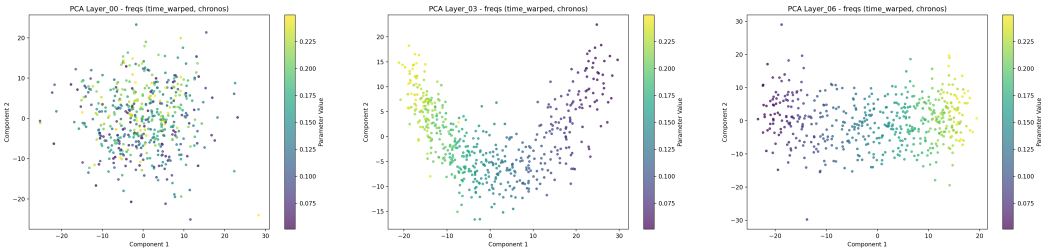


Figure 35: Time-Warped — Chronos — Freqs — PCA (Layers 00/03/06)

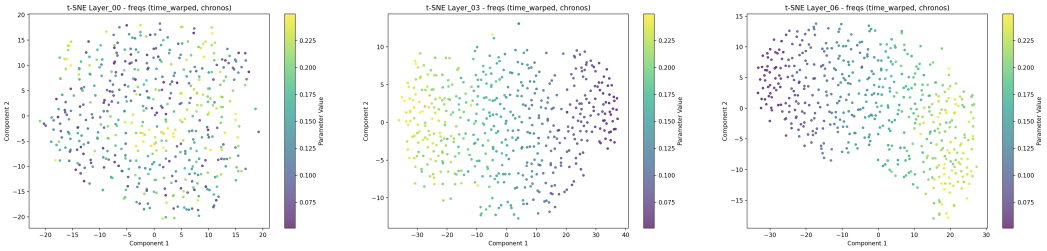


Figure 36: Time-Warped — Chronos — Freqs — t-SNE (Layers 00/03/06)

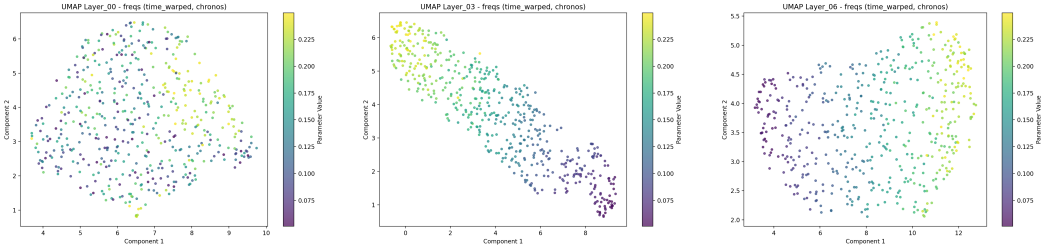


Figure 37: Time-Warped — Chronos — Freqs — UMAP (Layers 00/03/06)

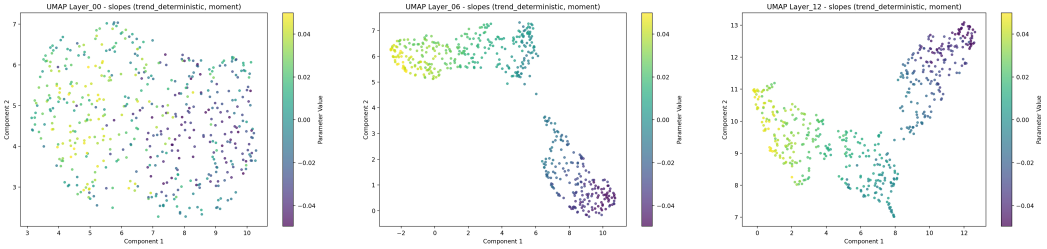


Figure 38: Trend — Moment — UMAP (Layers 00/06/12)

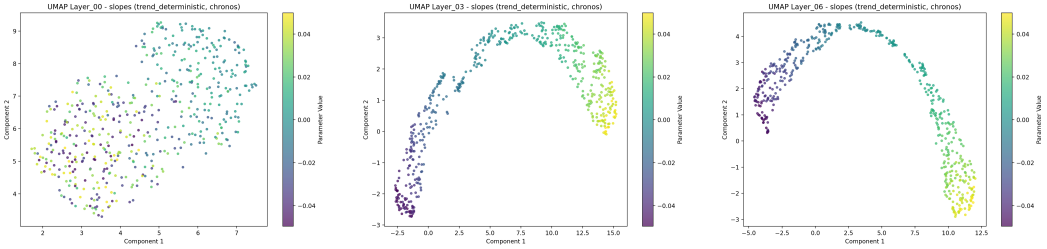


Figure 39: Trend — Chronos — UMAP (Layers 00/03/06)

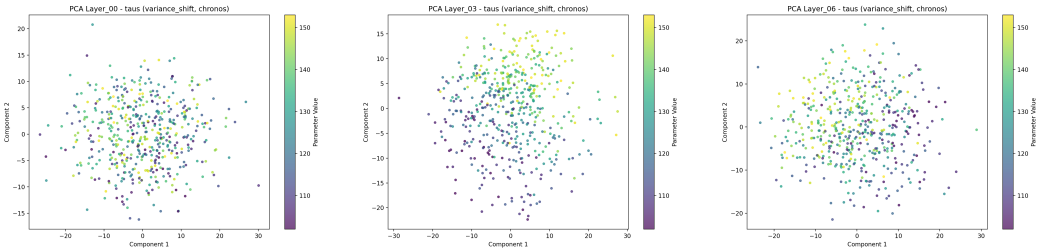


Figure 40: Variance Shift — Chronos — PCA (Layers 00/03/06)

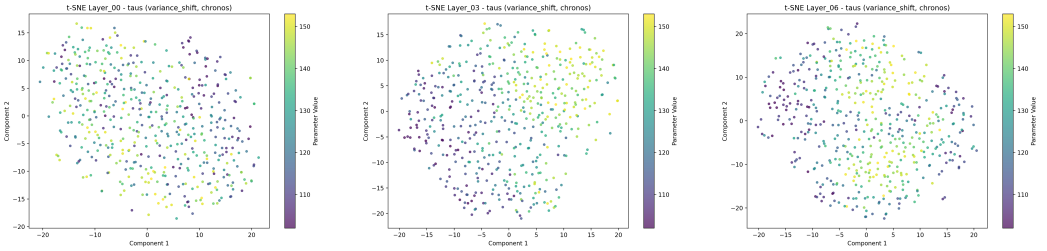


Figure 41: Variance Shift — Chronos — t-SNE (Layers 00/03/06)

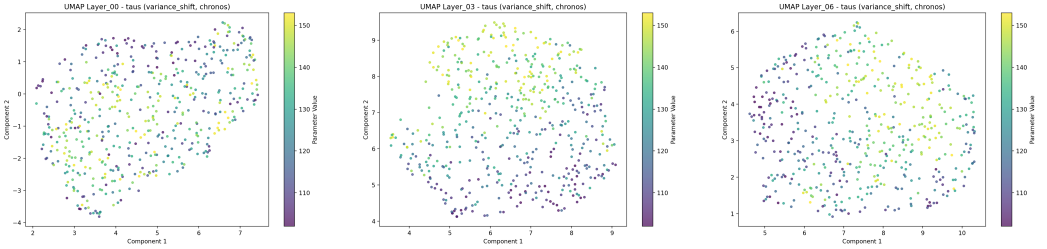


Figure 42: Variance Shift — Chronos — UMAP (Layers 00/03/06)

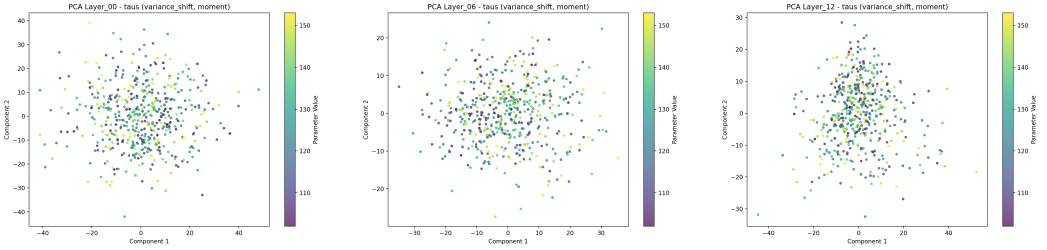


Figure 43: Variance Shift — Moment — PCA (Layers 00/06/12)

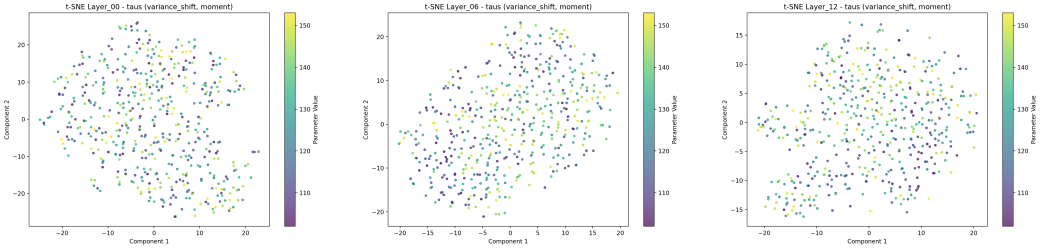


Figure 44: Variance Shift — Moment — t-SNE (Layers 00/06/12)

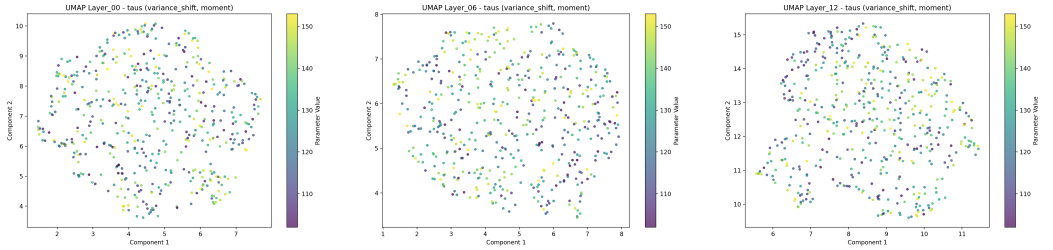
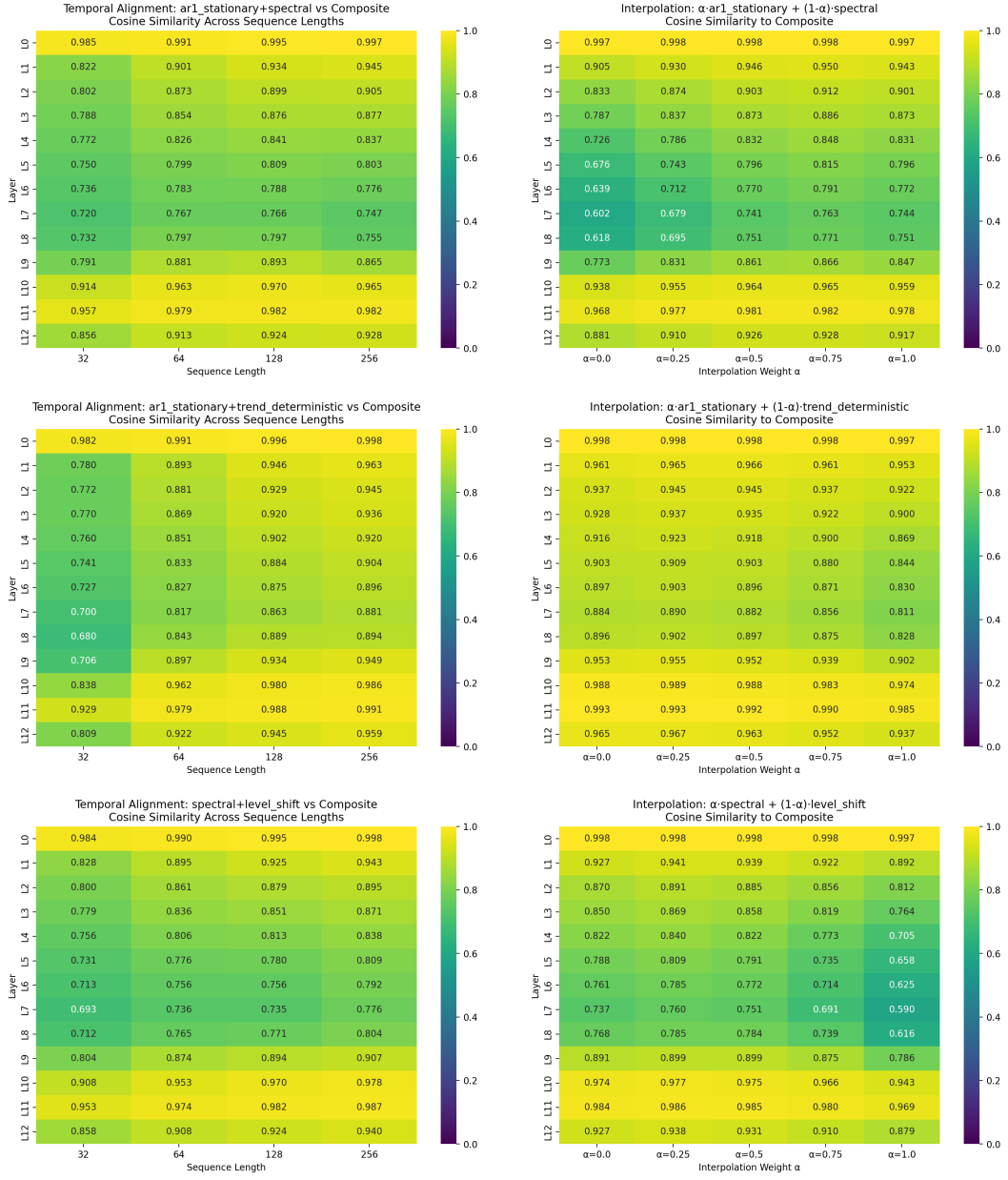
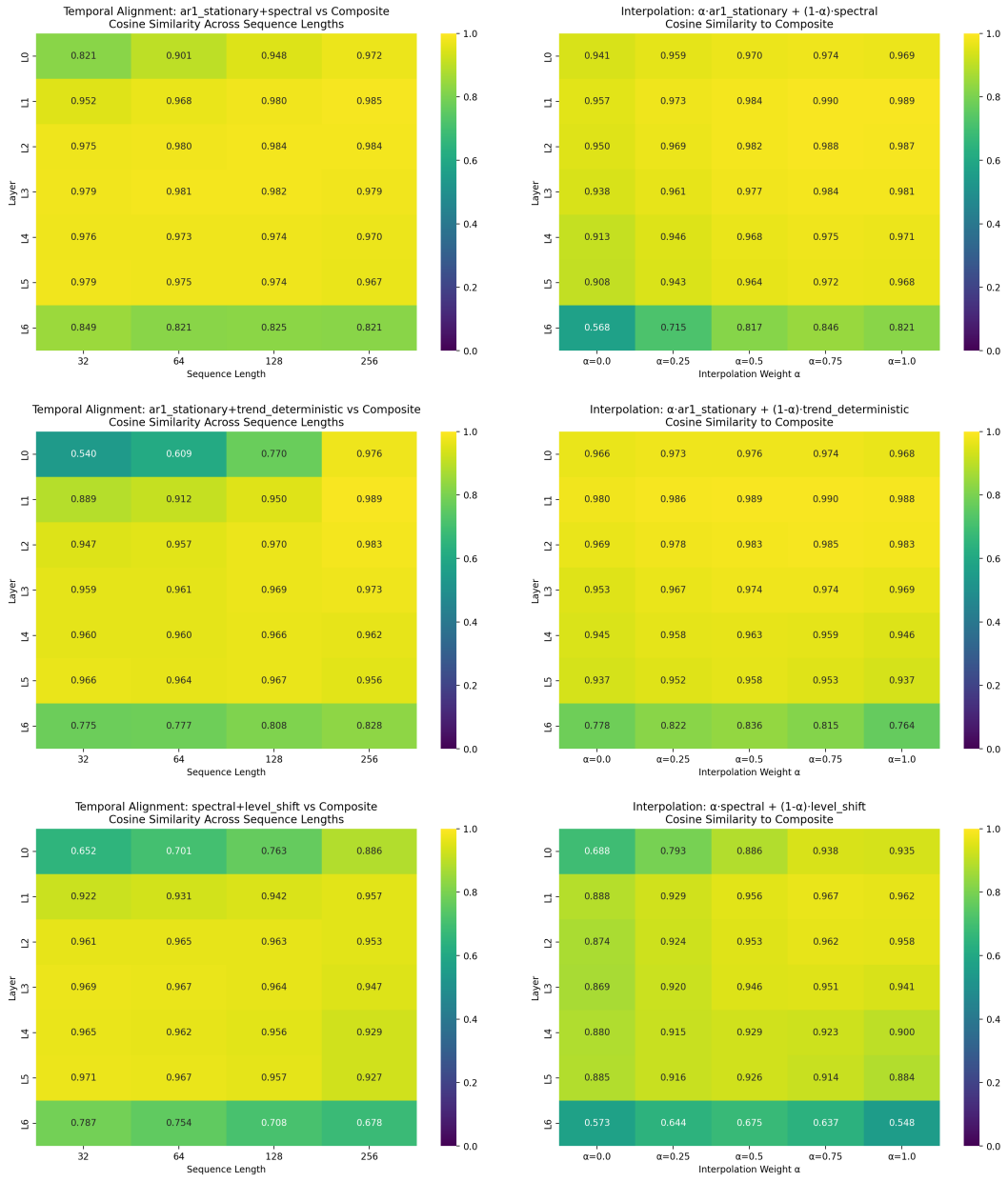


Figure 45: Variance Shift — Moment — UMAP (Layers 00/06/12)

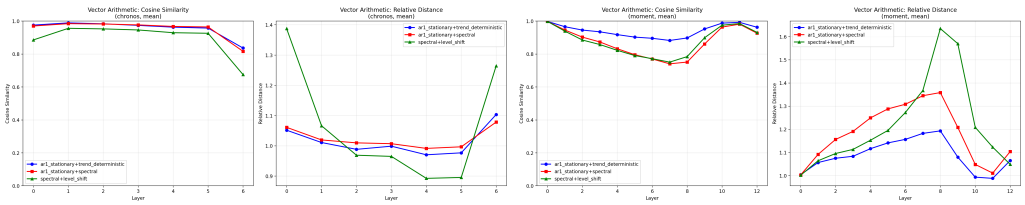


(a) MOMENT - Temporal alignment experiments

(b) MOMENT - Interpolation analysis



(a) Chronos - Temporal alignment experiments (b) Chronos - Interpolation analysis



(a) Vector Arithmetic Experiments with Chronos (b) Vector Arithmetic Experiments with MOMENT

Figure 48: Vector Arithmetic Experiments