Predictive Representation Learning for Partially Observed Neural Dynamics

Xinyi Li[1,*], **Zhichao Liang**[2,*], **Hongjun Jiang**[3], **Yian Zhu**[1], **Guanyi Zhao**[1], **Xuejian Yang**[1], **Xiaoqi Chen**[4], **Kexin Lou**[5], **Quanying Liu**[1,5,6,†]

[1] Department of Biomedical Engineering, Southern University of Science and Technology
[2] Center for Neurocognition and Social Behavior, Artificial Intelligence Research Institute, Shenzhen University of Advanced Technology
[3] Department of Artificial Intelligence, Southern University of Science and Technology
[4] School of Data Science, The Chinese University of Hong Kong, Shenzhen
[5] Omni-Intelligence, Shenzhen, China [6] Shenzhen Loop Area Institute

[*] These authors contributed equally. [†] Corresponding author: liuqy@sustech.edu.cn

Abstract

In long-term multi-session neural recordings, the same latent population dynamics are often observed through changing subsets of neural channels due to electrode drift, signal instability, or session-dependent recording variability. This creates a partially observed system-identification problem: each session provides only an incomplete view of a shared dynamical system, and missing channels are not merely absent inputs but unobserved variables coupled to the same latent dynamics. Existing approaches often rely on observation-pattern-specific mapping, which treats missing channels passively and leaves the shared latent dynamics weakly constrained. We propose CHAMA (CHannel-Aware Masked Attention), a mask-conditioned predictive representation learning framework for identifying shared neural dynamics under heterogeneous channel availability. CHAMA introduces a single global observation interface conditioned on channel-availability masks, allowing observed and unobserved channels to jointly constrain latent dynamics. The key idea is bidirectional: missing channels impose structural constraints on the latent representation, while the learned latent dynamics enable principled recovery of missing activity. To improve identifiability under sparse observations, CHAMA aggregates causal temporal windows, effectively trading temporal context for missing spatial information. Across synthetic dynamical systems and real multi-session neural recordings, CHAMA improves missing-channel completion, forecasting, latent dynamical fidelity, and downstream behavioral decoding over conditional adapter, zero-padding, and masked-autoencoding baselines. These results show that dynamics-aware recovery preserves globally consistent and behaviorally relevant structure beyond point-wise reconstruction accuracy. The source code is available at: https://github.com/ncclab-sustech/CHAMA.

1 Introduction

Prior work has shown that, under consistent task demands, neural population activity evolves on stable low-dimensional manifolds that persist across repeated behaviors and even across days [1–3]. Studying such shared dynamics requires tracking population activity across trials and over extended periods. Recent advances in large-scale neural recording make this possible [4, 5], enabling identification of shared dynamics from multi-session data. However, long-term recordings also introduce a critical complication: the observation interface is not fixed. Across recording sessions,

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

experimental setups, or transient sensor failures, only subsets of neural channels may be available, resulting in partially observed views of shared latent dynamics [6, 7]. The same neural process is measured through partial, time-varying observations rather than a fixed set of units (Fig. 1). These settings give rise to a partially observed system-identification problem, in which latent neural dynamics must be inferred from incomplete and time-varying observations.

Existing approaches often rely on observation-pattern-specific mapping. They assume a shared latent backbone while learning observation-pattern-specific read-in or readout adapters that map observations into a common latent space [3, 8–11]. This strategy can align neural activity across sessions, but it does not by itself solve the recoverability problem induced by partial observations. Because mappings depend on observed channels, absent channels provide no constraints on the latent-to-channel map, leaving shared dynamics underconstrained. Rather than being ignored or absorbed into conditional adapters, unobserved channels should be treated as missing but recoverable variables coupled to observed activity through shared dynamics [12, 13]. A well-identified latent dynamics model should therefore support reconstruction of both observed and unobserved activity [11, 14–16]. Failing to exploit this structure limits applications that require complete and consistent signal representations, such as longitudinal or closed-loop BCI systems [17, 18], and hinders scalability, as changes in channel availability may necessitate retraining or re-estimation of observation-pattern-specific mappings.

This calls for a unified observation interface that represents heterogeneous channel availability without introducing observation-pattern-specific mappings. In many BCI settings, consistent channel identities (e.g., EEG caps or implanted arrays) across trials provide useful structure that can be exploited. By conditioning on channel-availability masks, both observed and missing channels are incorporated into a common objective, enabling missing data to impose implicit constraints through shared latent dynamics. We implement this via **CHAMA**, a unified framework for handling heterogeneous missingness. CHAMA learns **a single global observation model conditioned on availability masks**. Beyond handling heterogeneous missingness, it leverages a key insight: **temporal redundancy can compensate for spatial sparsity**. Even when unobserved, a channel remains coupled to observed activity through shared dynamics. By temporal aggregation, the model can better infer the latent state and recover missing information, effectively trading time for information.

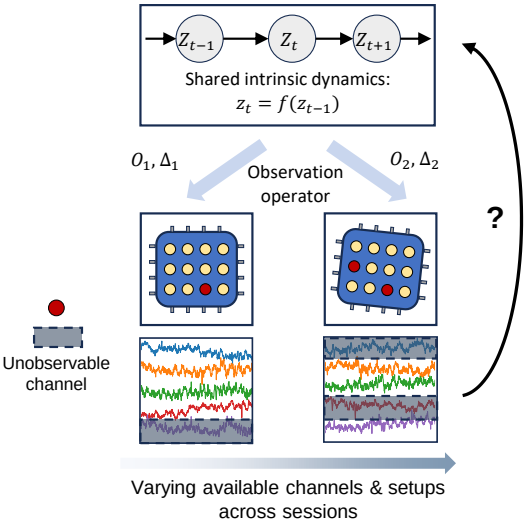

Figure 1: Shared intrinsic dynamics with varying observations.

We evaluate CHAMA on synthetic dynamical systems and real multi-session neural recordings with controlled missingness. Across benchmarks, our framework consistently outperforms conditional adapters, zero-padding baselines, and imputation methods without shared latent dynamics, in both missing-channel completion and forecasting. Beyond predictive accuracy, constraints from unobserved channels improve latent consistency, while the learned latent structure, in turn, supports inferring missing channels. Our contributions are as follows:

- We recast heterogeneous channel availability in multi-session neural recordings as a partially observed dynamical-system identification problem, in which missing channels are latent-coupled variables that can constrain shared dynamics rather than merely absent inputs.
- We introduce CHAMA, a unified observation interface that embeds all channels in a shared channel-identity space and conditions inference on availability masks, enabling full-channel prediction without conditional adapters.
- We provide theoretical and empirical evidence that temporal aggregation and missingness-aware objectives improve latent-state recoverability, yielding dynamically consistent and behaviorally useful reconstructions under heterogeneous partial observations.

2 Related Work

Partial Observations and Observability in Dynamical Systems. Our setting is naturally framed as learning a dynamical system from partial observations, a classical problem in system identification and control [19, 20]. A long line of work studies when latent states or dynamics are recoverable from incomplete measurements through notions of observability and identifiability [21–23]. Complementary theoretical perspectives such as delay-coordinate embeddings provide intuition that temporal information can compensate for limited instantaneous observability [24]. Takens' embedding theorem provides formal support for this idea by showing that the state can be reconstructed from a sufficiently long window of delayed measurements [25, 26]. Under suitable conditions, trajectories observed through a restricted measurement function can still encode the underlying state evolution [27, 28]. These results motivate trading time for information in partially observed systems and provide conceptual grounding for temporal aggregation in resolving missingness [29].

Data-Driven Neural System Identification and Cross-Session Alignment. Modern data-driven approaches increasingly learn dynamical models of neural systems directly from population activity using flexible function approximators and amortized inference [8, 30–33]. To reconcile recordings across days, multi-session methods typically assume a shared latent manifold with conditional readouts or learned adapters that map observations into a common space [3, 9]. While these shared-latent formulations effectively correct drift and support alignment, they typically treat missing channels as discarded observations rather than variables coupled to the latent dynamics. As a result, they fail to leverage the identifiability constraints induced by partially overlapping observations, leaving the latent dynamics underconstrained by any single condition's limited readout. Furthermore, because alignment relies on mappings tied to observation patterns, changes in channel availability alter the mapping, requiring re-learning and limiting generalization. Our method instead explicitly models the observation mask, allowing complementary partial observations across trials or sessions to jointly constrain the latent dynamics and enable inference of full-channel activity.

Representation Learning for Neural Time Series. Complementary to shared-latent dynamical models, masked autoencoder (MAE) approaches explicitly leverage missing channels as supervision, reconstructing masked patches to learn general-purpose representations [34, 35]. Diffusion-based models (e.g., CSDI, SSSD) achieve state-of-the-art imputation, but are computationally intensive and lack explicit causal latent states, limiting real-time low-latency BCI use [36, 37]. A separate line of work explicitly imputes missing channels using bidirectional RNNs or transformers [38, 39]. While such approaches minimize reconstruction error, they treat imputation as signal processing rather than system identification, failing to learn a coherent shared dynamical system. Consequently, they may produce locally plausible but globally inconsistent reconstructions, leading to temporal inconsistency, limited interpretability and reduced predictive utility. Moreover, they rely on reconstructing artificially masked data during training and typically employ random masking strategies for denoising or pre-training. In contrast, our setting involves *systematic, trial-long sensor dropout*, where missing channels lack ground truth and require identifying invariant dynamics rather than simple pattern completion.

Neural Interaction Modeling. Modeling interactions within neural populations provides insights into their underlying dynamics. Attention-based approaches, such as NetFormer, leverage transformer architectures to infer state-dependent inter-neuron connectivity [40], while graph-based methods capture dependencies among neural units through predefined or learned graph structures [41, 42]. By characterizing interactions across neural units, these approaches facilitate neural dynamics prediction and provide insights into the underlying network structure. These approaches demonstrate how modeling neural interactions can support dynamics prediction and reveal underlying network structure. However, leveraging cross-channel dependencies to recover shared latent dynamics under heterogeneous partial observations remains less explored.

3 Method

3.1 Problem Formulation

We consider a latent dynamical system evolving as

$$\mathbf{h}_{t+1} = f(\mathbf{h}_t), \quad \mathbf{h}_t \in \mathbb{R}^d, \tag{1}$$

where $\mathbf{h}_t$ is a low-dimensional state capturing shared neural dynamics and $f(\cdot)$ is an unknown transition function. We observe multichannel activity under structured, time-varying missingness across sessions and trials. The dataset consists of multiple sessions, each containing multiple trials. For trial r, observations follow

$$\mathbf{x}_t^{(r)} = M^{(r)} g^{(r)} \left(\mathbf{h}_t^{(r)} \right), \tag{2}$$

where $\mathbf{x}_t^{(r)} \in \mathbb{R}^C$ denotes the activity of C channels at time t in trial r, $M^{(r)} \in \{0,1\}^{C \times C}$ is a diagonal binary mask indicating the observed channels for that trial, and $g^{(r)}(\cdot)$ is a trial-dependent observation mapping that captures trial-specific recording variability and drift. We assume that the latent dynamics share a common structure across sessions, while allowing for trial-specific variability through the observation model. Moreover, every channel is observed in at least one trial, and channel identities are assumed to be aligned across sessions. Our goal is to learn (i) a predictive latent representation of shared dynamics from partially observed activity, and (ii) a unified observation interface for joint inference over observed and unobserved channels.

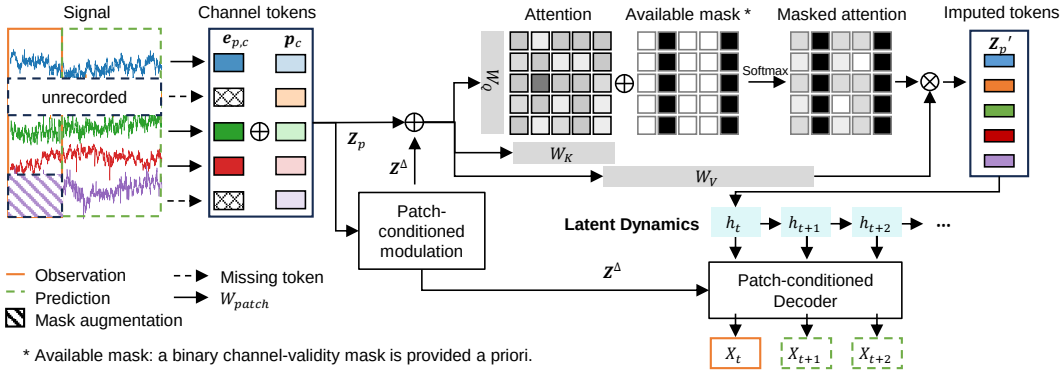

Figure 2: **CHAMA framework**. CHAMA learns a unified mask-conditioned observation model that adapts to heterogeneous missingness and nonstationary mappings. Masked attention aggregates information from available channels while incorporating missing ones into a shared objective. By leveraging temporal redundancy through latent dynamics, the model compensates for spatial sparsity and improves recovery of missing signals.

3.2 Channel-Aware Masked Attention

Patch tokenization with missing-channel token. Given a multichannel signal $\mathbf{X} \in \mathbb{R}^{C \times T}$ with C channels and length T, we extract unit-stride overlapping temporal patches of length L from each channel. Let $\mathbf{x}_{c,p} \in \mathbb{R}^L$ denote the p-th patch from channel c, where $p = 1, \ldots, T - L + 1$. Each patch is defined as $\mathbf{x}_{c,p} = \mathbf{X}_{c,\, p:\, p+L-1}$, such that the patch indexed by p is right-aligned at time step $t = p + L - 1$, and captures a local temporal window from the past for representing state at time t. Each patch $\mathbf{x}_{c,p}$ is mapped to an embedding $\mathbf{e}_{c,p} \in \mathbb{R}^d$ via a projection. To handle missing channels, we introduce a learnable missing embedding $\mathbf{m} \in \mathbb{R}^d$ and define

$$\tilde{\mathbf{e}}_{c,p} = \begin{cases} \mathbf{e}_{c,p}, & a_c = 1, \\ \mathbf{m}, & a_c = 0, \end{cases} \tag{3}$$

where $a_c \in \{0,1\}$ indicates channel availability. Rather than serving as a placeholder, the missing embedding represents an unobserved but dynamically coupled variable, allowing missing channels

to remain part of latent inference through shared dynamics. To encode channel identity, we add a learnable channel embedding $\mathbf{p}_c \in \mathbb{R}^d$:

$$\mathbf{z}_{c,p} = \tilde{\mathbf{e}}_{c,p} + \mathbf{p}_c. \tag{4}$$

The resulting channel-wise tokens form a matrix $\mathbf{Z}_p \in \mathbb{R}^{C\times d}$, which serves as input to the subsequent attention module.

Patch-conditioned modulation. We introduce a lightweight conditioning module that extracts a trial-specific embedding from early observations to disambiguate trial-dependent drift. Concretely, we use the token matrix from the first patch, $\mathbf{Z}_1 \in \mathbb{R}^{C\times d}$ as a compact and low-latency summary of the initial context, and map it through an MLP to produce a context embedding $\mathbf{Z}^{\Delta} \in \mathbb{R}^{C\times d}$. We then inject this context back into the token stream by residual addition:

$$\bar{\mathbf{Z}}_p = \mathbf{Z}_p + \mathbf{Z}^{\Delta}, \quad p = 1, \ldots, T - L + 1. \tag{5}$$

Masked self-attention over available channels. To enforce that only observable measurements contribute evidence, while preserving latent slots for unobserved variables, we restrict attention to available channels via a mask on the attention logits. For each patch p, we compute queries, keys, and values using learnable matrices $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d\times d}$:

$$\mathbf{Q}_p = \bar{\mathbf{Z}}_p\mathbf{W}_Q, \quad \mathbf{K}_p = \bar{\mathbf{Z}}_p\mathbf{W}_K, \quad \mathbf{V}_p = \bar{\mathbf{Z}}_p\mathbf{W}_V. \tag{6}$$

The attention logits are:

$$\mathbf{S}_p = \frac{\mathbf{Q}_p\mathbf{K}_p^{\top}}{\sqrt{d}} \in \mathbb{R}^{C\times C}. \tag{7}$$

To prevent attending to missing channels, we use an *availability mask* $\mathbf{M} \in \mathbb{R}^{C\times C}$ that masks *key columns*:

$$\mathbf{M}_{ij} = \begin{cases} 0, & a_j = 1, \\ -\infty, & a_j = 0, \end{cases} \tag{8}$$

where a_j indicates whether channel j is available. We then apply masked softmax:

$$\mathbf{A}_p = \mathrm{softmax}(\mathbf{S}_p + \mathbf{M}), \tag{9}$$

so unavailable channels receive zero attention weight. The updated channel tokens for patch p are computed as $\mathbf{Z}'_p = \mathbf{A}_p\mathbf{V}_p \in \mathbb{R}^{C\times d}$, yielding updated tokens $\mathbf{z}'_{c,p}$ for each channel c. Missing channels are excluded as evidence sources (keys), but remain active query slots whose representations are updated through interactions with observed channels and temporal context. This preserves the participation of unobserved variables in latent inference, enabling missing channels to impose indirect constraints on the shared dynamics.

3.3 Causal Latent Readout and Latent-space Dynamics

For each time step t, we use the patch that is right-aligned at t, and extract a latent state via a learnable readout function:

$$\mathbf{h}_t = \psi\big(\mathbf{Z}'_p\big), \quad p = t - L + 1. \tag{10}$$

Since $\mathbf{Z}'_p$ is computed from the causal window $\mathbf{X}_{:,\,t-L+1:t}$, $\mathbf{h}_t$ depends only on observations up to time t. From an observability perspective, the temporal patch acts as an approximate delay-coordinate embedding under partial measurements, where latent states become progressively more identifiable as additional temporal context is accumulated. Under suitable observability conditions, the resulting masked temporal embedding can locally determine the latent state despite incomplete instantaneous observations (Appendix B).

We model latent dynamics with an MLP predictor $\rho(\cdot)$ and enforce multi-step consistency by unrolling predictions for K steps:

$$\hat{\mathbf{h}}_{t+k+1} = \rho(\bar{\mathbf{h}}_{t+k}), \quad k = 0, \ldots, K - 1. \tag{11}$$

To mitigate error accumulation while still exposing the model to its own predictions, we apply periodic teacher forcing: every K_{tf} steps, the rollout is reset using the re-encoded ground-truth state.

$$\bar{\mathbf{h}}_\tau = \begin{cases} \hat{\mathbf{h}}_\tau, & \text{if } \tau \notin \mathcal{T}_{\mathrm{reset}}, \\ \mathbf{h}_\tau, & \text{if } \tau \in \mathcal{T}_{\mathrm{reset}}, \end{cases} \tag{12}$$

where $\mathcal{T}_{\mathrm{reset}} = \{t + K_{\mathrm{tf}},\, t + 2K_{\mathrm{tf}},\, \ldots\}$ within an unroll segment. This stabilizes long-horizon prediction while maintaining training on model-generated trajectories.

3.4 Patch-conditioned Decoder

We reconstruct observations from latent states using a conditional low-rank decoder:

$$\hat{\mathbf{X}}_{:,t} = \mathbf{W}_{\mathrm{dec}}(\mathbf{r})\, h_t + \mathbf{b}_{\mathrm{dec}}(\mathbf{r}). \tag{13}$$

The conditioning signal $\mathbf{r}$ is obtained by aggregating the context embedding $\mathbf{Z}^{\Delta}$ across the embedding dimension to produce a channel-wise summary (Appendix A). This allows the decoder to adapt its readout to channel configuration and trial-dependent variability, thereby improving reconstruction of missing-channel activity.

3.5 Training Objective with Mask Augmentation

Simulated missingness during training. To improve robustness to arbitrary missing-channel patterns, during training we additionally *hide* a subset of channels that are actually observed. For a trial r with observation mask $M^{(r)} \in \{0,1\}^{C\times C}$ (diagonal), we sample an auxiliary diagonal mask $A^{(r)} \in \{0,1\}^{C\times C}$, where $A^{(r)}_{cc} = 0$ indicates that channel c is artificially masked. We form the *effective* mask as $\tilde{M}^{(r)} = M^{(r)}A^{(r)}$, and run the model using $\tilde{M}^{(r)}$, i.e., channels with $\tilde{M}^{(r)}_{cc} = 0$ are treated as unavailable in attention and replaced by the missing embedding. When using $\tilde{M}^{(r)}$, artificially masked channels are excluded as inputs, but their ground-truth observations remain available for supervision. This encourages the model to infer masked channels from context and learn cross-channel dependencies.

Objectives in latent and signal space. We optimize a combination of (i) latent-space prediction loss, (ii) signal-space prediction loss, and (iii) reconstruction loss. Latent prediction is supervised by matching predicted latents to encoded latents:

$$\mathcal{L}_{\text{lat}} = \sum_{\tau} \left\| \hat{\mathbf{h}}_{\tau} - \mathbf{h}_{\tau} \right\|_2^2. \tag{14}$$

To leverage supervision from channels that are *observed in the dataset* (including those artificially hidden by $A^{(r)}$), we compute signal-space losses only on the originally observed channels indicated by $M^{(r)}$ (not $\tilde{M}^{(r)}$). Specifically, with decoder $d(\cdot)$,

$$\mathcal{L}_{\text{sig-pred}} = \sum_{\tau} \left\| M^{(r)} \Big(d(\hat{\mathbf{h}}_{\tau}) - \mathbf{X}_{:,\tau} \Big) \right\|_2^2. \tag{15}$$

$$\mathcal{L}_{\text{rec}} = \sum_{t=L}^{T} \left\| M^{(r)} (d(\mathbf{h}_t) - \mathbf{X}_{:,t}) \right\|_2^2. \tag{16}$$

The overall training objective is

$$\mathcal{L} = \mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{sig-pred}} + \mathcal{L}_{\text{rec}}. \tag{17}$$

4 Experiments

Datasets. We evaluate our method on five benchmarks. **Benchmarks 1 and 2** are synthetic datasets generated from the Lorenz system [43], where the ground-truth latent dynamics are known. **Benchmark 1 (B1)** represents an information-limited setting. The three Lorenz channels provide complementary, non-redundant information about the underlying dynamics. **Benchmark 2 (B2)** introduces redundancy resembling population-level recordings and includes session-wise drift. Masking is shared across trials within a session, introducing structured, persistent missingness that mimics stable channel dropout and aligns with session-wise drift. This design tests whether cross-channel redundancy can compensate for consistently missing channels under known ground truth. **Benchmarks 3 and 4 (B3 and B4)** use publicly available datasets containing simultaneously recorded motor and premotor cortex activity from macaques, along with corresponding behavioral outputs. The two benchmarks differ in their behavioral tasks: B3 uses the center-out (CO) reaching task [44], whereas B4 uses the maze task [45]. **Benchmark 5 (B5)** uses the human iBCI dataset [3], comprising

motor cortical neural activity recorded from a participant with tetraplegia performing an imagined-movement cursor control task. Unlike the synthetic benchmarks, the ground-truth latent dynamics of all three real neural recording datasets are unknown. To reflect realistic recording conditions, we apply trial-wise random masking, simulating transient missingness commonly observed in neural data. Data generation and preprocessing details are provided in Appendix C.

Baselines. We compare our method against three types of baselines that reflect common strategies for handling missing channels and cross-trial variability. (i) **Conditional adapter** baseline fits a lightweight, observation-pattern-specific channel adapter that maps each pattern's observations into a shared latent space. This paradigm is widely adopted for handling cross-session variability [3, 8–10]. In B1, the absence of session-wise observation drift justifies the use of a **shared readout** to decode latent states. In B2, however, **session-specific readouts** are required to model observation variability, which in turn prevents evaluating how latent dynamics constrain the recovery of unobserved channels. (ii) **Zero-padding** baseline replaces missing channels with zeros and evaluates the model on the resulting fixed-dimensional inputs, serving as a simple reference that tests whether gains arise from principled recovery rather than naive imputation. (iii) We additionally compare against a strong imputation baseline (**NeuroPaint** [39]) based on an MAE framework. As this approach focuses on reconstructing missing observations, our comparison primarily evaluates the quality of the imputed signals, while highlighting the importance of modeling latent dynamics beyond observation-level reconstruction. Implementation details of the baselines and CHAMA are provided in Appendices F and H.

Evaluation Metrics. We assess performance at both the observational level and the level of latent dynamics. At the observational level, we report explained variance (EV) of reconstructed signals on unobserved channels defined by the observation mask $M^{(s)}$. During training, additional simulated missingness is introduced via an auxiliary mask $\tilde{M}^{(s)}$, but evaluation is performed only with respect to the observational missingness. At the latent dynamical level, for **B1** and **B2**, we assess predictive EV to measure how well the inferred dynamics capture temporal evolution. Also, we quantify dynamical fidelity via the R^2 of a linear readout from inferred latents to the true underlying dynamics (see Appendix G). For **B3**, **B4** and **B5**, where latent dynamics are unknown, we evaluate the quality of inferred representations by linearly decoding the behavioral variable (2D cursor/hand velocity) from both the latent space and the reconstructed signal space.

4.1 Non-redundant partial observations of Lorenz dynamics

In Benchmark 1, observations are non-redundant, so each observed channel provides limited information about the missing ones. Despite this setting, CHAMA consistently improves unobserved-channel reconstruction and long-horizon prediction over all baselines (Fig. 3d–g, i–l). The inferred latent trajectories also achieve higher latent fidelity, indicating more accurate recovery of the underlying dynamics. The proposed temporal patching mechanism further compensates for the information deficiency caused by partial and non-redundant observations (Fig. 3h). By aggregating temporal context, longer observation windows provide stronger constraints on the latent state and underlying dynamics, improving their identifiability and reducing ambiguity in inferring missing signals. Beyond the model design itself, we also observe that the data composition plays an important role. As shown in Exp.2 (Fig. 3m), fully observed samples anchor the model to the complete state space, enabling better identification of cross-channel dependencies and latent dynamics. This leads to a more robust mapping from partial observations to the full latent state and improves generalization to OOD missing patterns. Ablation analysis on **B1** further demonstrates that both the proposed modules and loss functions are critical to CHAMA's robustness and overall performance (Appendix Tables 1 and 2).

4.2 Redundant observations with session drift of Lorenz dynamics

Benchmark 2 introduces redundant observations together with session-wise drift, allowing us to evaluate whether models can exploit cross-channel redundancy under changing observation mappings. As shown in Fig. 4b, dynamical models exhibit clear advantages over MAE-based NeuroPaint in imputing unobservable channels. This suggests that incorporating dynamical structure allows the model to utilize information more globally and coherently across time, rather than relying solely on local observation patterns. Since NeuroPaint does not model latent dynamics or support prediction, we restrict the following analysis to models with explicit dynamical latent spaces. While

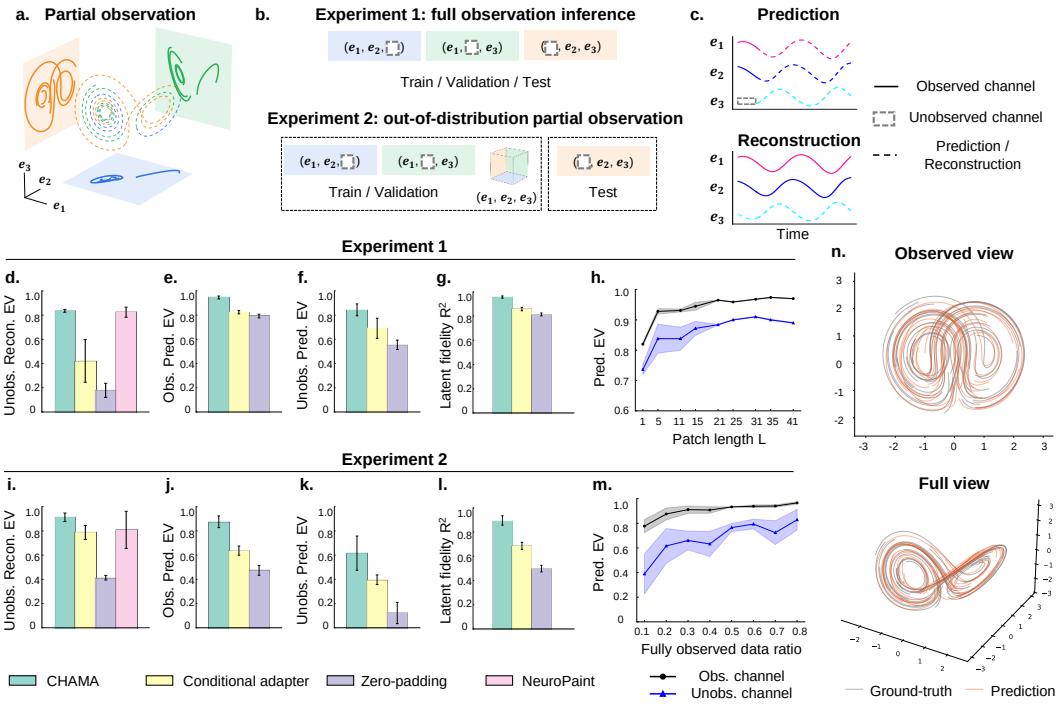

Figure 3: **Non-redundant partial observations of Lorenz dynamics.** (a) Partial observation setup. (b) Different distribution over missingness patterns in two experiments. (c) Task formulation: long-horizon prediction, using the initial temporal patch to predict the full future sequence of all channels; and cross-channel reconstruction, inferring the unobserved channels from the observed ones. **(d–h) Exp. 1:** CHAMA consistently outperforms baselines in unobserved channel reconstruction and full-channel prediction. It also achieves superior latent representation fidelity, which is computed based on the predicted latent trajectories, reflecting more accurate recovery of the underlying dynamics. Longer temporal windows provide richer dynamical cues and reduce uncertainty in latent state estimation. **(i–m) Exp. 2:** Similar trends to Exp. 1 are observed, demonstrating the robustness and generalizability of CHAMA. A higher proportion of fully observed data in the training set significantly improves prediction performance on unobserved channels. (n) Predicted trajectories match the ground truth and recover the characteristic Lorenz "butterfly" attractor in 3D. **Note**: NeuroPaint focuses on reconstructing missing observations, not prediction.

the conditional adapter baseline performs well on observed channel prediction, the latent fidelity remains substantially worse (Fig. 4e), suggesting that fitting only observable channels is insufficient for recovering consistent latent dynamics. Beyond relative performance, we also examine how performance evolves as the problem becomes more ill-posed (Fig. 4f). Increasing the proportion of unobserved channels reduces constraints on the latent state, making inference and reconstruction more difficult. At the same time, compared with the non-redundant setting, temporal aggregation provides smaller gains under redundancy because the latent state can already be inferred from subsets of channels (Fig. 4g). Nevertheless, ablation analysis shows that the proposed modules remain important, suggesting that robust recovery of latent dynamics still requires leveraging both dynamical structure and complementary constraints across channels and time (Appendix Table 3).

4.3 Partial observations in experimental neural dynamics

For Benchmarks 3–5, we exclude conditional adapters, as trial-specific random masking introduces a combinatorial space of missing-channel patterns, making pattern-specific adaptation computationally impractical and difficult to generalize. Fig. 5 reveals a notable discrepancy between neural signal reconstruction quality and downstream behavioral decoding performance across three experimental datasets. Although the MAE-based method NeuroPaint achieves reconstruction accuracy comparable

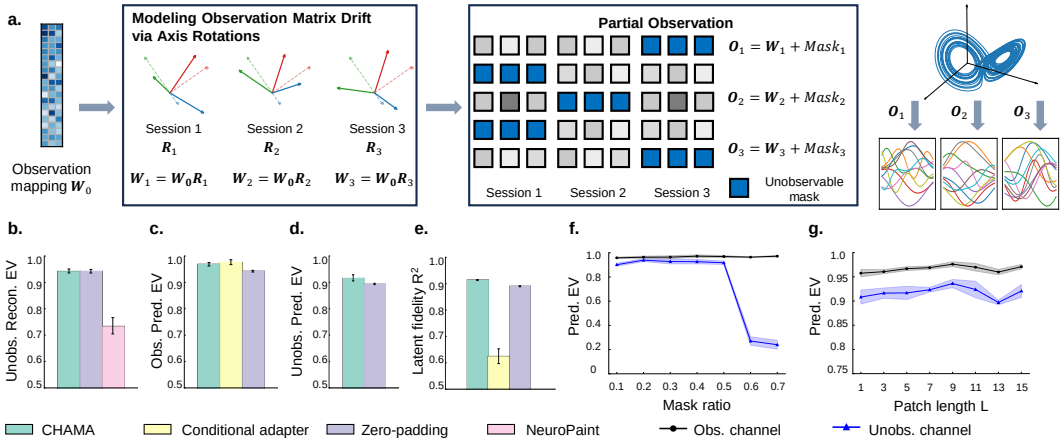

Figure 4: **Redundant partial observations with session drift of Lorenz dynamics.** (a) Observation mapping drift is modeled via session-specific rotations of a base mapping, yielding multiple observation matrices. (b) Models based on latent dynamics significantly outperform the MAE-based reconstruction model (NeuroPaint) on unobserved channel reconstruction. (c) All methods achieve near-ceiling performance on observed channel prediction, likely due to redundancy. (d) For unobserved channel prediction, CHAMA consistently outperforms Zero-padding by a small margin. (e) Without constraints from unobserved channels, the latent representations learned by the conditional adapter degrade. (f) Predictive performance across different proportions of unobserved channels. (g) Predictive performance across different patch lengths.

to CHAMA (Fig. 5b, f, j), this does not translate into equally strong decoding performance. As the mask ratio increases, decoding performance generally declines across all methods, whereas CHAMA consistently achieves competitive or superior performance when decoding from reconstructed signals (Fig. 5c, g, k). Similar trends are observed when decoding from inferred latent states (Fig. 5d, h, l), although the improvements are more modest. These results suggest that point-wise reconstruction accuracy alone does not fully reflect the preservation of behaviorally relevant neural information. By explicitly modeling latent neural dynamics, CHAMA may better retain the temporal structure important for downstream decoding, even when its reconstruction accuracy is comparable to that of MAE-based approaches. The consistent trends across macaque reaching tasks and human iBCI recordings further support the applicability of CHAMA under partial neural observability.

5 Discussion and Conclusion

In this work, we consider a common yet underexplored setting in long-term neural recordings, where shared latent dynamics are observed through nonstationary observation interfaces. We propose **CHAMA**, a mask-conditioned predictive representation learning framework. Our results highlight several key insights into learning dynamical systems under partial and heterogeneous observations. First, jointly modeling unobserved channels as variables coupled through shared dynamics imposes stronger structural constraints on the latent space, leading to improved identifiability and more consistent dynamics compared to approaches that ignore missing channels. Second, under channel redundancy, modeling latent dynamics better exploits shared information across channels and time, yielding more coherent representations than reconstruction-based methods. Third, dynamical inference preserves temporally consistent structure beyond pointwise reconstruction accuracy, leading to improved downstream decoding. Finally, the benefit of temporal aggregation depends on the structure of cross-channel information: in non-redundant settings, longer temporal windows significantly improve state identifiability by trading time for information, whereas in redundant regimes, the marginal gain from temporal context is limited.

Despite these advantages, several limitations suggest directions for future work. In particular, it assumes the existence of shared latent dynamics across sessions and aligned channel identities, which may not hold under strong nonstationarities or cross-modal settings. An important direction for future

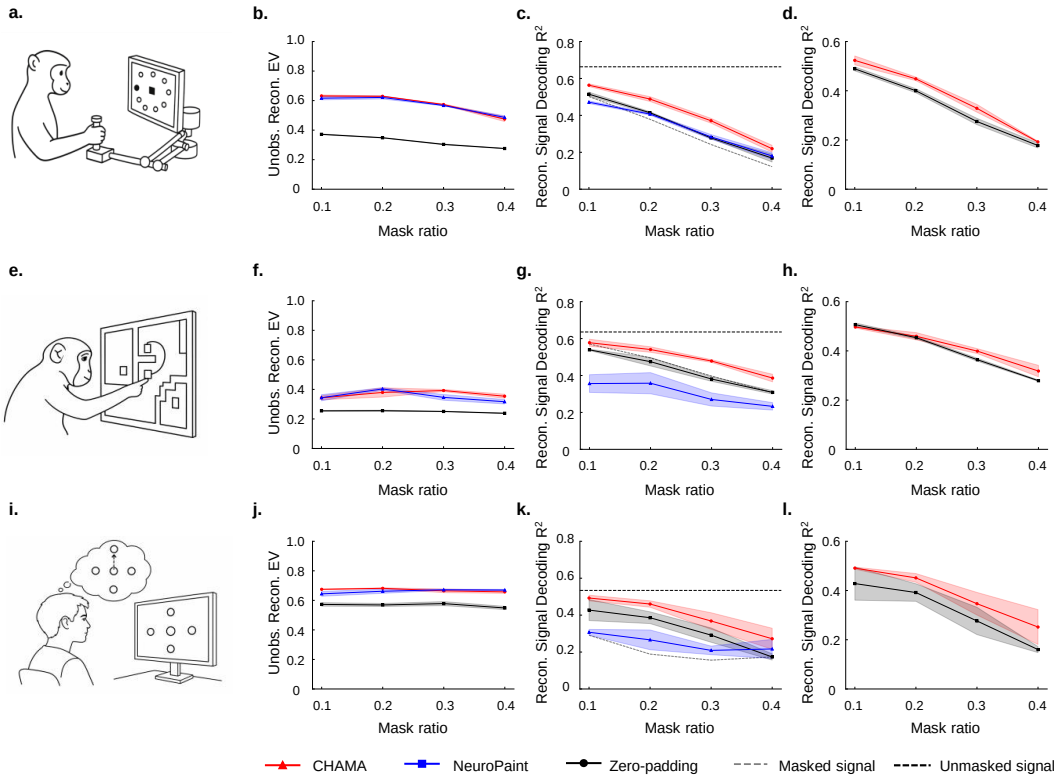

Figure 5: **Partial observations in experimental neural recording data.** Experimental setups for three neural recording benchmarks: (a) macaque center-out reaching task (B3), (e) macaque maze task (B4), and (i) human imagined-movement cursor control task (B5). (b, f, j) Unobserved channel reconstruction performance across different mask ratios, where the mask ratio denotes the fraction of neural recording channels randomly masked per trial. (c, g, k) Behavioral decoding performance using reconstructed neural signals, compared with decoding from masked and unmasked signals. (d, h, l) Behavioral decoding performance using inferred latent states.

work is to combine CHAMA with a learned alignment step (e.g., coarse region-level alignment), enabling broader applicability when exact cross-session identities are unavailable. Moreover, recovery of missing channels fundamentally depends on the observability of the latent state from partially observed windows. When this condition is violated, accurate reconstruction is information-theoretically impossible.

Acknowledgments and Disclosure of Funding

This work was supported by National Natural Science Foundation of China (3254100307, 62472206), Brain Science and Brain-like Intelligence Technology-National Science and Technology Major Project (2021ZD0200500, 2026ZD0220402), National Key R&D Program of China (2025YFC3410000), Shenzhen Science and Technology Innovation Committee (RCYX20231211090405003, JCYJ20240813141105007), Guangdong Basic and Applied Basic Research Foundation (2026B1515020099, 2025A1515011645), Guangdong S&T Program (2026B0101110003), Shanghai Municipal Special Program for Basic Research on General AI Foundation Models (2025SHZDZX026D05), Shenzhen Loop Area Institute (FPF10120250012), and the open research fund of the Guangdong Provincial Key Laboratory of Mathematical and Neural Dynamical Systems, the Center for Computational Science and Engineering at Southern University of Science and Technology, Shenzhen Key Laboratory of Smart Healthcare Engineering.

References

[1] Juan A Gallego, Matthew G Perich, Stephanie N Naufel, Christian Ethier, Sara A Solla, and Lee E Miller. Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. *Nature communications*, 9(1):4233, 2018.

[2] Juan A Gallego, Matthew G Perich, Raeed H Chowdhury, Sara A Solla, and Lee E Miller. Long-term stability of cortical population dynamics underlying consistent behavior. *Nature neuroscience*, 23(2):260–270, 2020.

[3] Brianna M Karpowicz, Yahia H Ali, Lahiru N Wimalasena, Andrew R Sedler, Mohammad Reza Keshtkaran, Kevin Bodkin, Xuan Ma, Daniel B Rubin, Ziv M Williams, Sydney S Cash, et al. Stabilizing brain-computer interfaces through alignment of latent dynamics. *Nature Communications*, 16(1):4662, 2025.

[4] Michael Okun, Armin Lak, Matteo Carandini, and Kenneth D Harris. Long term recordings with immobile silicon probes in the mouse cortex. *PloS one*, 11(3):e0151180, 2016.

[5] Nicholas A Steinmetz, Cagatay Aydin, Anna Lebedeva, Michael Okun, Marius Pachitariu, Marius Bauza, Maxime Beau, Jai Bhagat, Claudia Böhm, Martijn Broux, et al. Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings. *Science*, 372(6539): eabf4588, 2021.

[6] János A Perge, Mark L Homer, Wasim Q Malik, Sydney Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Intra-day signal instabilities affect decoding performance in an intracortical neural interface system. *Journal of neural engineering*, 10(3): 036004, 2013.

[7] Max Dabagia, Konrad P Kording, and Eva L Dyer. Aligning latent representations of neural activity. *Nature Biomedical Engineering*, 7(4):337–343, 2023.

[8] Chethan Pandarinath, Daniel J O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky, Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg, et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature methods*, 15(10):805–815, 2018.

[9] Ayesha Vermani, Il Memming Park, and Josue Nassar. Leveraging generative models for unsupervised alignment of neural time series data. In *The Twelfth International Conference on Learning Representations*, 2023.

[10] Noga Mudrik, Ryan Ly, Oliver Ruebel, and Adam Shabti Charles. Creimbo: Cross-regional ensemble interactions in multi-view brain observations. In *The Thirteenth International Conference on Learning Representations*.

[11] Mehdi Azabou, Vinam Arora, Venkataramana Ganesh, Ximeng Mao, Santosh Nachimuthu, Michael Mendelson, Blake Richards, Matthew Perich, Guillaume Lajoie, and Eva Dyer. A unified, scalable framework for neural population decoding. *Advances in Neural Information Processing Systems*, 36:44937–44956, 2023.

[12] Srini Turaga, Lars Buesing, Adam M Packer, Henry Dalgleish, Noah Pettit, Michael Hausser, and Jakob H Macke. Inferring neural population dynamics from multiple partial recordings of the same neural circuit. *Advances in Neural Information Processing Systems*, 26, 2013.

[13] Felix Pei, Joel Ye, David Zoltowski, Anqi Wu, Raeed H Chowdhury, Hansem Sohn, Joseph E O'Doherty, Krishna V Shenoy, Matthew T Kaufman, Mark Churchland, et al. Neural latents benchmark '21: Evaluating latent variable models of neural population activity. *Advances in Neural Information Processing Systems*, 2021.

[14] Peter Y Lu, Joan Ariño Bernad, and Marin Soljačić. Discovering sparse interpretable dynamics from partial observations. *Communications Physics*, 5(1):206, 2022.

[15] Yizi Zhang, Yanchen Wang, Donato M Jiménez-Benetó, Zixuan Wang, Mehdi Azabou, Blake Richards, Renee Tung, Olivier Winter, Eva Dyer, Liam Paninski, et al. Towards a" universal translator" for neural dynamics at single-cell, single-spike resolution. *Advances in Neural Information Processing Systems*, 37:80495–80521, 2024.

[16] Christopher Versteeg, Andrew R Sedler, Jonathan D McCart, and Chethan Pandarinath. Expressive dynamics models with nonlinear injective readouts enable reliable recovery of latent features from neural activity. *ArXiv*, pages arXiv–2309, 2023.

[17] Jordan L Vasko, Laura Aume, Sanjay Tamrakar, Samuel C IV Colachis, Collin F Dunlap, Adam Rich, Eric C Meyers, David Gabrieli, and David A Friedenberg. Increasing robustness of brain–computer interfaces through automatic detection and removal of corrupted input signals. *Frontiers in Neuroscience*, 16:858377, 2022.

[18] Yun-Ting Kuo, Han-Lin Wang, Bo-Wei Chen, Ching-Fu Wang, Yu-Chun Lo, Sheng-Huang Lin, Po-Chuan Chen, and You-Yin Chen. Degradation-aware neural imputation: Advancing decoding stability in brain machine interfaces. *APL bioengineering*, 9(2), 2025.

[19] Andreas Schlaginhaufen, Philippe Wenk, Andreas Krause, and Florian Dorfler. Learning stable deep dynamics models for partially observed or delayed dynamical systems. *Advances in Neural Information Processing Systems*, 34:11870–11882, 2021.

[20] Hanna Kurniawati. Partially observable markov decision processes and robotics. *Annual review of control, robotics, and autonomous systems*, 5(1):253–277, 2022.

[21] Robert Hermann and Arthur Krener. Nonlinear controllability and observability. *IEEE Transactions on automatic control*, 22(5):728–740, 2003.

[22] William Qian, Jacob A Zavatone-Veth, Benjamin S Ruben, and Cengiz Pehlevan. Partial observation can induce mechanistic mismatches in data-constrained models of neural dynamics. *Advances in Neural Information Processing Systems*, 37:67467–67510, 2024.

[23] Anming Gu, Edward Chien, and Kristjan Greenewald. Partially observed trajectory inference using optimal transport and a dynamics prior. In *International Conference on Learning Representations*, 2025.

[24] Charles D Young and Michael D Graham. Deep learning delay coordinate dynamics for chaotic attractors from partial observable data. *Physical Review E*, 107(3):034215, 2023.

[25] Floris Takens. Detecting strange attractors in turbulence. *Lecture Notes in Mathematics*, 898: 366–381, 1981.

[26] Jonah Botvinick-Greenhouse, Robert Martin, and Yunan Yang. Invariant measures in time-delay coordinates for unique dynamical system identification. *Physical Review Letters*, 135(16): 167202, 2025.

[27] Joseph Bakarji, Kathleen Champion, J Nathan Kutz, and Steven L Brunton. Discovering governing equations from partial measurements with deep delay autoencoders. *Proceedings of the Royal Society A*, 479(2276):20230422, 2023.

[28] Hao Peng, Pei Chen, Rui Liu, and Luonan Chen. Spatiotemporal information conversion machine for time-series forecasting. *Fundamental Research*, 4(6):1674–1687, 2024.

[29] Genevieve Flaspohler, Nicholas A Roy, and John W Fisher III. Belief-dependent macro-action discovery in pomdps using the value of information. *Advances in Neural Information Processing Systems*, 33:11108–11118, 2020.

[30] Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Advances in neural information processing systems*, 21, 2008.

[31] Joshua Glaser, Matthew Whiteway, John P Cunningham, Liam Paninski, and Scott Linderman. Recurrent switching dynamical systems models for multiple interacting neural populations. *Advances in neural information processing systems*, 33:14867–14878, 2020.

[32] Pedro Herrero-Vidal, Dmitry Rinberg, and Cristina Savin. Across-animal odor decoding by probabilistic manifold alignment. *Advances in Neural Information Processing Systems*, 34: 20360–20372, 2021.

[33] Arthur Pellegrino, N Alex Cayco Gajic, and Angus Chadwick. Low tensor rank learning of neural dynamics. *Advances in Neural Information Processing Systems*, 36:11674–11702, 2023.

[34] Zhe Li, Zhongwen Rao, Lujia Pan, Pengyun Wang, and Zenglin Xu. Ti-mae: Self-supervised masked time series autoencoders. *arXiv preprint arXiv:2301.08871*, 2023.

[35] Jiaxiang Dong, Haixu Wu, Haoran Zhang, Li Zhang, Jianmin Wang, and Mingsheng Long. SimMTM: A simple pre-training framework for masked time-series modeling. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[36] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in neural information processing systems*, 34:24804–24816, 2021.

[37] Juan Lopez Alcaraz and Nils Strodthoff. Diffusion-based time series imputation and forecasting with structured state space models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

[38] Sabera J Talukder, Jennifer J. Sun, Matthew K Leonard, Bingni W Brunton, and Yisong Yue. Deep neural imputation: A framework for recovering incomplete brain recordings. In *NeurIPS 2022 Workshop on Learning from Time Series for Health*, 2022.

[39] Ji Xia, Yizi Zhang, Shuqi Wang, Genevera I. Allen, Liam Paninski, Cole Lincoln Hurwitz, and Kenneth D. Miller. Inpainting the neural picture: Inferring unrecorded brain area dynamics from multi-animal datasets. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[40] Ziyu Lu, Wuwei Zhang, Trung Le, Hao Wang, Uygar Sümbül, Eric SheaBrown, and Lu Mi. Netformer: An interpretable model for recovering dynamical connectivity in neuronal population dynamics. In *International Conference on Learning Representations*, volume 2025, pages 99987–100015, 2025.

[41] Jingyuan Li, Leo Scholl, Trung Le, Pavithra Rajeswaran, Amy Orsborn, and Eli Shlizerman. Amag: Additive, multiplicative and adaptive graph neural network for forecasting neuron activity. *Advances in Neural Information Processing Systems*, 36:8988–9014, 2023.

[42] Kijung Yoon. Self-supervised discovery of neural circuits in spatially patterned neural responses with graph neural networks. *Advances in Neural Information Processing Systems*, 38:144893–144920, 2026.

[43] Edward N. Lorenz. Deterministic nonperiodic flow. *Journal of Atmospheric Sciences*, 20(2): 130 – 141, 1963. doi: 10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2.

[44] Xuan Ma, Fabio Rizzoglio, Kevin L Bodkin, Eric Perreault, Lee E Miller, and Ann Kennedy. Using adversarial networks to extend brain computer interface decoding accuracy over time. *eLife*, 12:e84296, aug 2023. ISSN 2050-084X.

[45] Felix Pei, Joel Ye, David Zoltowski, David Zoltowski, Anqi Wu, Raeed Chowdhury, Hansem Sohn, Joseph O' Doherty, Krishna V Shenoy, Matthew Kaufman, Mark Churchland, Mehrdad Jazayeri, Lee Miller, Jonathan Pillow, Il Memming Park, Eva Dyer, and Chethan Pandarinath. Neural latents benchmark '21: Evaluating latent variable models of neural population activity. In J. Vanschoren and S. Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021.

A Patch-conditioned decoder details

To obtain the conditioning signal $\mathbf{r}$, we summarize the patch-conditioned modulation $\mathbf{Z}^{\Delta} \in \mathbb{R}^{C\times d}$ by averaging over the embedding dimension:

$$\mathbf{r}_c = \frac{1}{d}\sum_{k=1}^{d} \mathbf{Z}^{\Delta}_{c,k}, \quad c = 1, \ldots, C, \tag{18}$$

resulting in a channel-wise summary vector $\mathbf{r} \in \mathbb{R}^C$ that captures trial-specific characteristics.

We reconstruct observations from latent states using a trial-conditioned decoder based on a low-rank adaptation of a shared linear readout. Given a latent state $\mathbf{h}_t$, the reconstruction is defined as

$$\hat{\mathbf{X}}_{:,t} = d(\mathbf{h}_t; \mathbf{r}) = (\mathbf{W}_0 + \Delta\mathbf{W}(\mathbf{r}))\,\mathbf{h}_t + \mathbf{b}_0 + \Delta\mathbf{b}(\mathbf{r}), \tag{19}$$

where $\mathbf{W}_0$ and $\mathbf{b}_0$ are shared base decoder parameters, and $\mathbf{r}$ denotes the trial-specific conditioning signal.

Instead of generating a full decoder weight matrix directly, we use a LoRA-style low-rank modulation:

$$\Delta\mathbf{W}(\mathbf{r}) = \mathbf{U}\,\mathrm{diag}(\mathbf{s}(\mathbf{r}))\,\mathbf{V}, \tag{20}$$

where $\mathbf{U} \in \mathbb{R}^{C\times r}$ and $\mathbf{V} \in \mathbb{R}^{r\times d_h}$ are shared low-rank factors, r is the adaptation rank, and $\mathbf{s}(\mathbf{r}) \in \mathbb{R}^r$ is a trial-dependent scaling vector generated from the trial conditioning signal. The bias shift is similarly generated as

$$\Delta\mathbf{b}(\mathbf{r}) = \phi_b(\mathbf{r}). \tag{21}$$

Equivalently, the decoder output can be written as

$$\hat{\mathbf{X}}_{:,t} = \mathbf{W}_0\mathbf{h}_t + \mathbf{b}_0 + \mathbf{U}\,(\mathbf{s}(\mathbf{r}) \odot \mathbf{V}\mathbf{h}_t) + \Delta\mathbf{b}(\mathbf{r}), \tag{22}$$

where $\odot$ denotes element-wise multiplication.

In our implementation, $\mathbf{s}(\mathbf{r})$ and $\Delta\mathbf{b}(\mathbf{r})$ are produced by lightweight linear hypernetworks. Their final layers are initialized to zero, so the decoder initially reduces to the shared base decoder:

$$d(\mathbf{h}_t; \mathbf{r}) \approx \mathbf{W}_0\mathbf{h}_t + \mathbf{b}_0. \tag{23}$$

This initialization stabilizes training while allowing trial-specific low-rank corrections to be learned gradually.

B Theoretical Analysis: Recoverability under Partial Observations

In this section, we provide a more explicit characterization of when latent states and missing channels are recoverable under partial observations, and explain why temporal aggregation improves identifiability. We further relate these results to the design of CHAMA.

B.1 Linear Observability under Partial Masking

We first consider the linear dynamical system

$$h_{t+1} = Ah_t, \tag{24}$$
$$x_t = Ch_t, \tag{25}$$

where $h_t \in \mathbb{R}^{d_h}$ is the latent state and $x_t \in \mathbb{R}^C$ is the full observation.

Under partial observations, only a subset of channels is visible. Let

$$P = \mathrm{diag}(m), \quad m \in \{0,1\}^C, \tag{26}$$

denote the observation mask. The observed signal becomes

$$\mathbf{y}_t = PCh_t. \tag{27}$$

We define the masked observability matrix over a temporal window of length L:

$$\mathcal{O}_L = \begin{bmatrix} PC \\ PCA \\ PCA^2 \\ \vdots \\ PCA^{L-1} \end{bmatrix}. \tag{28}$$

Theorem 1 (Recoverability under Partial Observations)**.** *Assume the latent dynamics are deterministic and linear. The latent state h_t is uniquely recoverable from a masked observation window*

$$(y_t, y_{t+1}, \ldots, y_{t+L-1})$$

if and only if

$$\operatorname{rank}(\mathcal{O}_L) = d_h. \tag{29}$$

Consequently, all channels, including unobserved channels, are recoverable through

$$x_t = Ch_t.$$

Proof. Expanding the dynamics recursively yields

$$y_t = PCh_t, \tag{30}$$
$$y_{t+1} = PCAh_t, \tag{31}$$
$$y_{t+2} = PCA^2h_t, \tag{32}$$
$$\vdots \tag{33}$$

which gives

$$\begin{bmatrix} y_t \\ y_{t+1} \\ \vdots \\ y_{t+L-1} \end{bmatrix} = \mathcal{O}_L h_t. \tag{34}$$

The latent state h_t is uniquely determined if and only if the linear system admits a unique solution, i.e.,

$$\operatorname{rank}(\mathcal{O}_L) = d_h.$$

Once h_t is recovered, all channels follow directly from

$$x_t = Ch_t.$$

□

B.2 Temporal Aggregation Trades Time for Information

Theorem 1 formalizes the intuition that temporal context can compensate for missing spatial observations.

Suppose the effective observed dimension per time step is

$$d_o = \operatorname{rank}(PC).$$

A necessary condition for recoverability is

$$L \cdot d_o \geq d_h. \tag{35}$$

Thus, when fewer channels are observed (smaller d_o), longer temporal windows are required to recover the latent state. Equivalently,

$$L \geq \left\lceil \frac{d_h}{d_o} \right\rceil. \tag{36}$$

This result provides a quantitative interpretation of the tradeoff between temporal and spatial information: increasing the temporal window length improves recoverability by accumulating complementary dynamical constraints over time.

B.3 Extension to Nonlinear Systems

We now consider the nonlinear system

$$h_{t+1} = f(h_t), \tag{37}$$
$$x_t = g(h_t), \tag{38}$$

where f and g are continuously differentiable.

Define the masked delay embedding map

$$\Phi_L(h) = \big(Pg(h), Pg(f(h)), \ldots, Pg(f^{L-1}(h))\big). \tag{39}$$

Theorem 2 (Local Recoverability in Nonlinear Systems)**.** *If the Jacobian of Φ_L satisfies*

$$\operatorname{rank}\left(D\Phi_L(h^\star)\right) = d_h \tag{40}$$

at the true latent state $h^\star$, then the latent state is locally identifiable from the masked temporal window.

Proof. By the inverse function theorem, full-rank Jacobian implies that Φ_L is locally injective around $h^\star$. Therefore, the latent state is uniquely determined within a local neighborhood by the masked observation window. □

This condition can be interpreted as a nonlinear analogue of observability. As the temporal window length increases, the delay embedding accumulates additional dynamical constraints, making local injectivity increasingly likely under generic dynamics.

B.4 When Recovery Is Impossible

Non-identifiability. If Φ_L is not locally injective at $\mathbf{h}^\star$, then there exist distinct states $\mathbf{h}^{(1)} \neq \mathbf{h}^{(2)}$ such that

$$\Phi_L(\mathbf{h}^{(1)}) = \Phi_L(\mathbf{h}^{(2)}).$$

Any estimator based solely on the observed window must produce identical predictions for both states. If these states yield different current channel values, recovery is information-theoretically impossible.

Unobservable latent directions. A sufficient condition for non-injectivity is the existence of a curve $\gamma(s)$ such that

$$Pg\big(f^{(k)}(\gamma(s))\big) \equiv \text{constant} \quad \text{for all } k = 0, \ldots, L-1.$$

Along such directions, the observations remain unchanged while the latent state varies, making certain channels unrecoverable.

B.5 Why a Mask-conditioned Dynamical Model Improves Identifiability

We now explain why a shared mask-conditioned dynamical model imposes stronger identifiability constraints than observation-pattern-specific adapters.

Setup. Consider two observation patterns (or sessions) with different partial observations of the same latent dynamical system:

$$h_{t+1} = f(h_t), \tag{41}$$
$$x_t^{(1)} = P_1 g(h_t), \tag{42}$$
$$x_t^{(2)} = P_2 g(h_t), \tag{43}$$

where P_1 and P_2 are distinct masking operators.

Observation-pattern-specific approaches typically learn separate encoders

$$z_t^{(1)} = E_1(x_t^{(1)}), \tag{44}$$
$$z_t^{(2)} = E_2(x_t^{(2)}), \tag{45}$$

followed by reconstruction losses defined only on the observed channels.

Non-identifiability of independently aligned latent spaces. Suppose the latent representation is transformed by any invertible mapping

$$\tilde{z}^{(i)} = T_i(z^{(i)}),$$

where T_i may differ across observation patterns.

If reconstruction is evaluated only on observed channels, then each encoder-decoder pair can absorb the transformation independently:

$$x_t^{(i)} \approx D_i(T_i^{-1}\tilde{z}_t^{(i)}). \tag{46}$$

Therefore, the latent coordinate systems across observation patterns are not uniquely constrained. In particular, there is no requirement that

$$T_1 = T_2.$$

As a result, multiple incompatible latent geometries can produce identical reconstruction performance on observed channels.

Shared dynamics impose cross-pattern consistency. In CHAMA, all observation patterns share a common latent dynamics model:

$$\hat{h}_{t+1} = \rho(h_t). \tag{47}$$

The latent trajectories inferred from different masking patterns must therefore satisfy the same transition operator ρ.

Suppose two observation patterns use incompatible latent transformations:

$$\tilde{h}^{(1)} = T_1(h), \quad \tilde{h}^{(2)} = T_2(h),$$

with $T_1 \neq T_2$.

For both transformed systems to obey the same latent dynamics, we would require

$$T_1(f(h)) = \rho(T_1(h)), \tag{48}$$
$$T_2(f(h)) = \rho(T_2(h)). \tag{49}$$

Equivalently,

$$\rho = T_1 \circ f \circ T_1^{-1} = T_2 \circ f \circ T_2^{-1}. \tag{50}$$

This condition generally fails unless the transformations are dynamically equivalent. Consequently, incompatible latent coordinate systems cannot simultaneously satisfy a shared latent dynamics model.

Effect of missing-channel supervision. CHAMA further introduces reconstruction losses on artificially masked channels:

$$\mathcal{L}_{\text{rec}} = \sum_t \|M(d(h_t) - x_t)\|_2^2. \tag{51}$$

Thus, the latent state must contain sufficient information not only to reconstruct currently observed channels, but also to infer channels that are temporarily unobserved.

This introduces additional constraints coupling different observation patterns through the shared latent manifold.

Together, shared latent dynamics and mask-conditioned reconstruction reduce the space of admissible latent representations. Rather than learning observation-pattern-specific embeddings that only explain local measurements, the model is forced to learn latent trajectories that are globally dynamically consistent across heterogeneous partial observations. This provides a theoretical explanation for why CHAMA improves latent identifiability and missing-channel recovery under structured partial observability.

B.6 Implications for CHAMA

The above analysis directly motivates the design of CHAMA.

Temporal patch aggregation. CHAMA encodes latent states from temporal patches rather than instantaneous observations. Theorems 1 and 2 suggest that increasing the temporal window length improves latent-state identifiability under partial masking by accumulating dynamical evidence over time.

Mask-conditioned latent inference. CHAMA explicitly conditions attention on the observation mask. Observed channels contribute evidence through attention keys and values, while missing channels remain active latent query slots. This preserves latent variables corresponding to unobserved channels and allows information to propagate through shared dynamics.

Why conditional adapters are insufficient. Observation-pattern-specific adapters learn separate mappings for each masking configuration. However, when observation patterns have limited overlap, the corresponding latent spaces become weakly constrained relative to one another. In contrast, CHAMA uses a single shared latent dynamics model conditioned on masks, forcing all partial observations to lie on the same dynamical manifold. This introduces additional identifiability constraints that improve recovery of missing channels.

B.7 Discussion

Our analysis suggests that missing-channel recovery is fundamentally an observability problem. Recovery becomes possible when the masked temporal window uniquely constrains the latent state. The role of CHAMA is therefore not merely to reconstruct signals, but to learn a shared latent representation that maximizes dynamical consistency across heterogeneous partial observations. These results also clarify the role of temporal aggregation: longer windows improve recoverability by trading temporal information for missing spatial information. This effect is strongest in non-redundant observation regimes, where instantaneous observations alone are insufficient to determine the latent state.

C Dataset

C.1 Benchmark 1

Partial Observations of the Lorenz System. The famous three-dimensional Lorenz attractor [43] is widely used as a benchmark in data-driven system research. Originally derived as a low-dimensional model of atmospheric convection, the Lorenz system exhibits sensitive dependence on initial conditions and chaotic dynamics. Its trajectories are bounded yet aperiodic, evolving on a fractal invariant set with the characteristic "butterfly-shaped" geometry, which makes long-horizon prediction and state reconstruction from partial observations particularly challenging. Its governing equations are

$$\begin{aligned}\dot{x} &= \sigma(y - x),\\ \dot{y} &= x(\rho - z) - y,\\ \dot{z} &= xy - \beta z.\end{aligned}$$

In our experiments, we generate multivariate trajectory segments from the three-dimensional Lorenz system with fixed parameters ($\sigma = 10,\ \rho = 48,\ \beta = 3.5$), and a fixed initial condition $(x_0, y_0, z_0) = (1.0,\ 0.0,\ 20.0)$. We numerically integrate the ODE using an explicit Euler scheme with time step $\Delta t = 1/500$, producing a long discrete-time trajectory. Also, we apply channel-wise z-score normalization to the observed sequences for each dimension. We then partition this trajectory into contiguous non-overlapping segments of length $T = 250$. The resulting collection of segments is then divided into disjoint training, validation, and test splits. Specifically, we allocate a subset of segments for model fitting (train), a separate subset for early stopping (validation), and hold out the remaining segments for final evaluation (test). The splits are constructed at the segment level, so that no time steps from the same segment appear in multiple splits, ensuring a fair assessment of generalization performance.

For **Experiment 1**, all segments are partially observed. Concretely, for each split (train/validation/test), we mask one channel for one third of the segments: one third are masked in

the x-channel, one third in the y-channel, and the remaining one third in the z-channel. As a result, every segment contains missing values in exactly one state dimension, and the overall dataset consists exclusively of partial observations.

For **Experiment 2**, the training and validation splits contain a mixture of fully and partially observed segments. Specifically, a proportion of segments are kept fully observed, while the remaining segments are partially observed with two-dimensional views, split evenly between the (x, y) and (y, z) channels (i.e., masking z for half of the partial segments and masking x for the other half). We treat the full observation proportion as a controllable parameter and report its impact in Sec. 4.1. In contrast, the test split is entirely composed of (x, z) observations, i.e., all test segments have the y-channel masked, creating a distribution shift between training/validation and testing.

C.2 Benchmark 2

Lorenz Data Processed by Rotation, Projection and Masking. Starting from the simulated Lorenz trajectories, we further construct a redundant, session-dependent observation space via a linear mapping. Let $X \in \mathbb{R}^{B\times T\times 3}$ denote the latent Lorenz states, where B is the number of trajectory segments and $T = 150$ is the segment length. We generate observed features $Y \in \mathbb{R}^{B\times T\times d_{\text{out}}}$ (with $d_{\text{out}} = 20$) by

$$Y_{b,t} = W_{s(b)} X_{b,t}, \qquad W_s \in \mathbb{R}^{d_{\text{out}}\times 3},$$

where $s(b) \in \{0, \dots, S-1\}$ is the session index assigned to segment b, and $S = 5$ is the total number of sessions.

Base mapping. We first sample a base matrix $W_0 \in \mathbb{R}^{d_{\text{out}}\times 3}$ from a Gaussian distribution and normalize it row-wise. In our implementation, we use a softmax normalization across the three input dimensions so that each row forms a simplex weight vector:

$$\tilde{W}_0 \sim \mathcal{N}(0, I), \qquad W_0(i,:) = \mathrm{softmax}\Big(\tilde{W}_0(i,:)\Big).$$

This produces nonnegative mixing weights and ensures comparable row scales across channels.

Session-wise drift via rotations. To model random changes in the observation mechanism across sessions, we construct a rotation matrix $R_s \in \mathbb{R}^{3\times 3}$ for each session using Rodrigues' formula: given a unit rotation axis $a = (a_x, a_y, a_z)^\top \in \mathbb{R}^3$ and a rotation angle $\theta \in \mathbb{R}$, the corresponding rotation matrix $R_s(a, \theta) \in \mathbb{R}^{3\times 3}$ can be written in closed form by Rodrigues' formula:

$$R_s(a, \theta) = I_3 \cos\theta + (1 - \cos\theta)\, aa^\top + \sin\theta\, [a]_\times,$$

where I_3 is the 3×3 identity matrix and $[a]_\times$ is the skew-symmetric matrix

$$[a]_\times = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix}.$$

In our implementation, we first normalize the axis $a \leftarrow a/\|a\|_2$ and then compute $R_s(a, \theta)$ using the above expression.

θ_s is sampled uniformly from $[-\theta_{\max}, \theta_{\max}]$, where $\theta_{\max} = 8°$. The session-specific observation matrices are obtained by right-multiplying the base mapping:

$$W_s = W_0 R_s.$$

As a result, each session applies a slightly different linear mixing of the latent Lorenz coordinates, creating a controlled distribution shift in the observation space across sessions.

Session-wise standardization. After mapping, we normalize the observed signals by z-scoring within each session. Concretely, for each session s and each output channel c, we compute the mean and standard deviation over all segments and time steps belonging to that session and channel, and apply

$$X_{b,t,c} = \frac{Y_{b,t,c} - \mu_{s(b),c}}{\sigma_{s(b),c} + \varepsilon}, \tag{52}$$

where ε is a small constant for numerical stability. In our experiments we use per-session, per-channel normalization, i.e., each session-channel pair has its own $(\mu_{s,c}, \sigma_{s,c})$. This removes session-dependent offsets and amplitude variations independently for each channel, without enforcing shared scaling across channels.

C.3 Benchmark 3

Dataset description. We used a center-out reaching dataset [44] in which a non-human primate performed repeated reaches from a central start position to one of multiple peripheral targets while neural activity was recorded with an intracortical microelectrode array. Neural data were represented as threshold-crossing events obtained from high-pass filtered broadband recordings and binned in time. Behavioral measurements were derived from the task cursor kinematics recorded at high sampling rate and synchronized to the neural stream. The dataset therefore consists of paired neural population activity and continuous kinematic trajectories over many trials and sessions, enabling evaluation of decoding and stability across time. We preprocessed the center-out data following the procedure described in previous studies [3].

Binning and resampling. Neural events were first binned at 1 ms resolution to obtain per-channel spike counts. To match the temporal resolution used downstream, spike counts were aggregated by summing within each target bin (20 ms bins). For continuous behavioral signals (cursor velocity), we downsampled to the target bin width by averaging within each bin, and resampled the signals to the same rate as the neural data. Also, we smoothed the binned spike trains using a Gaussian kernel (50 ms width). To reduce the influence of atypical behavior on training and evaluation, we identified and removed outliers in the continuous behavioral variables.

Behavioral Data Normalization Behavioral data were preprocessed by first subtracting the mean of each of the two behavioral dimensions. The mean-centered data were then normalized by the maximum absolute value across both dimensions, scaling the values to the range ([-1, 1]). In addition, target labels were converted into one-hot encoded vectors for subsequent model input.

C.4 Benchmark 4

Dataset description. We used the maze dataset from the Neural Latents Benchmark [13], containing neural recordings from the primary motor cortex (M1) and dorsal premotor cortex (PMd) of a macaque monkey performing an instructed-delay reaching task. During the task, the monkey reached toward visually presented targets while avoiding virtual barriers, producing diverse hand movement trajectories across different task conditions. The dataset includes simultaneously recorded neural spiking activity and hand kinematics.

Binning and resampling. We aligned each trial to the go cue and extracted a 1.3 s window following the cue. Neural spike trains were discretized into non-overlapping 20 ms bins and smoothed along the temporal dimension using a Gaussian kernel with a standard deviation of 80 ms. Two-dimensional hand velocities were averaged within the corresponding time bins to ensure temporal alignment with neural activity.

Behavioral Data Normalization Behavioral data were preprocessed by first subtracting the mean of each of the two behavioral dimensions, computed across all samples. The mean-centered data were then standardized by dividing each dimension by its standard deviation, with a small constant (10^{-8}) added for numerical stability.

C.5 Benchmark 5

Dataset description. We used the human intracortical brain-computer interface (iBCI) dataset [3]. The participant had tetraplegia resulting from a cervical spinal cord injury and was implanted with two 96-channel microelectrode arrays in the left precentral gyrus, providing recordings from 192 intracortical channels. During the experiment, the participant performed a closed-loop cursor control task by imagining right-hand movements toward four peripheral targets. We excluded calibration trials and retained only successful trials. We extracted the binned neural activity, two-dimensional cursor velocity and target position, along with their corresponding timestamps, for subsequent analyses.

Binning and resampling. We processed the neural recordings by binning threshold-crossing events into non-overlapping 20 ms time bins, yielding spike-count sequences with a temporal resolution of 50 Hz. To reduce high-frequency fluctuations and obtain smoother estimates of neural activity, the

binned spike counts were convolved along the temporal dimension with a Gaussian kernel with a width of 10 ms. The resulting smoothed neural activity was used as input for subsequent analyses.

Behavioral Data Normalization Behavioral data were preprocessed by first subtracting the mean of each of the two behavioral dimensions. The mean-centered data were then normalized by the maximum absolute value across both dimensions, scaling the values to the range ([-1, 1]). In addition, target labels were converted into one-hot encoded vectors for subsequent model input.

D Hyperparameter Analysis

The effect of K_{tf} is shown in Appendix Fig. 1. As K_{tf} increases, prediction performance improves and eventually saturates, indicating that longer teacher forcing primarily enhances the model's ability to capture *global, long-horizon dynamics*. With small K_{tf}, training is dominated by short rollout segments, which can lead to dynamics that are locally consistent but drift over longer horizons. Increasing K_{tf} effectively exposes the model to extended trajectories, encouraging alignment with the underlying long-term evolution and reducing accumulated prediction errors. This effect is more pronounced for the unobserved channel, where accurate prediction depends on correctly inferring latent state transitions. The progressive improvement with larger K_{tf} suggests that longer conditioning provides stronger constraints on latent dynamics, leading to more coherent and stable representations. In contrast, the weak dependence of reconstruction performance on K_{tf} implies that reconstruction is largely governed by instantaneous mappings rather than temporal consistency. Overall, these results suggest that the primary role of increasing K_{tf} is to mitigate short-horizon bias and promote the learning of globally consistent dynamics, particularly in the presence of latent structure.

We further investigate the sensitivity of CHAMA to augmented channel masking and conditioning module capacity, as well as its robustness to cross-session channel misalignment, on Benchmark 2. The results are summarized in Appendix Fig. 2. During training, we do not use a fixed number of additionally masked channels. Instead, the masking ratio is randomly sampled in a range, exposing the model to diverse missingness patterns. As shown in Appendix Fig. 2(a), prediction performance remains relatively stable at moderate masking ratios, whereas excessive masking leads to noticeable degradation, particularly on unobserved channels. This suggests that sufficient channel information is necessary to reliably capture cross-channel dependencies.

The patch-conditioned modulation module provides trial-level context, while the shared latent transition model is responsible for capturing temporal dynamics through multi-step prediction objectives. Appendix Fig. 2(b) shows that varying the hidden size of the modulation MLP has little effect on prediction performance. This weak sensitivity suggests that the conditioning module does not simply explain away the temporal dynamics. The required conditioning capacity may depend on the complexity of the underlying drift. In Benchmark 2, where the drift is static and low-dimensional, a relatively small modulation network appears sufficient.

CHAMA assumes correspondence between nominal channels across sessions, which is natural for electrode-level recordings such as LFP or MUA, where electrode locations remain identifiable even though the contributing neuronal populations may drift. To examine robustness to violations of this assumption, we introduce session-specific rotations with angles sampled from ($[-\theta_{\max}, \theta_{\max}]$). These rotations alter the effective population readout associated with each nominal channel, with larger ($\theta_{\max}$) inducing greater cross-session mismatch. As shown in Appendix Fig. 2(c), prediction performance decreases gradually with increasing misalignment, especially for unobserved channels, while retaining reasonable performance even under substantial perturbations. These results suggest that CHAMA can tolerate moderate deviations from exact channel alignment, although stable channel correspondence remains beneficial.

E Ablation Study

In this section, we present the full ablation results for Benchmarks 1–5. We evaluate the contribution of each architectural component by removing it individually and measuring the resulting performance across different experimental settings. The ablation results (Appendix Tab. 1–6) reveal that the

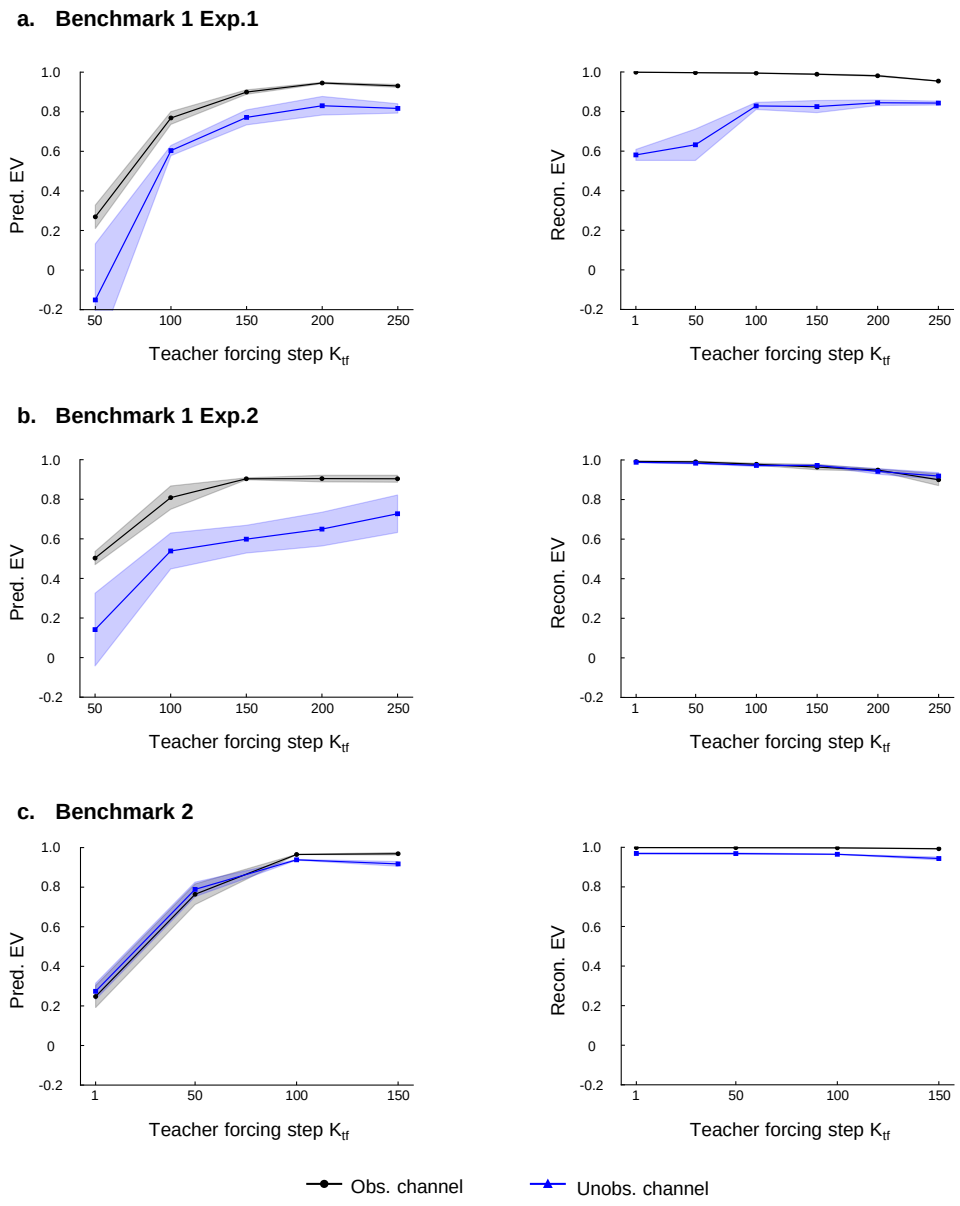

Figure 1: **Prediction and reconstruction performance under varying teacher forcing steps K_{tf} across benchmarks.** (a) Benchmark 1 Exp.1; (b) Benchmark 1 Exp.2 (fully observed data ratio = 0.2); (c) Benchmark 2 (mask ratio = 0.5). The left column shows prediction explained variance (Pred. EV), and the right column shows reconstruction explained variance (Recon. EV). Black curves denote the observed channel (Obs. channel), and blue curves denote the unobserved channel (Unobs. channel). Shaded regions indicate standard deviation. As K_{tf} increases, prediction performance generally improves and saturates, while reconstruction performance remains high and relatively stable across settings.

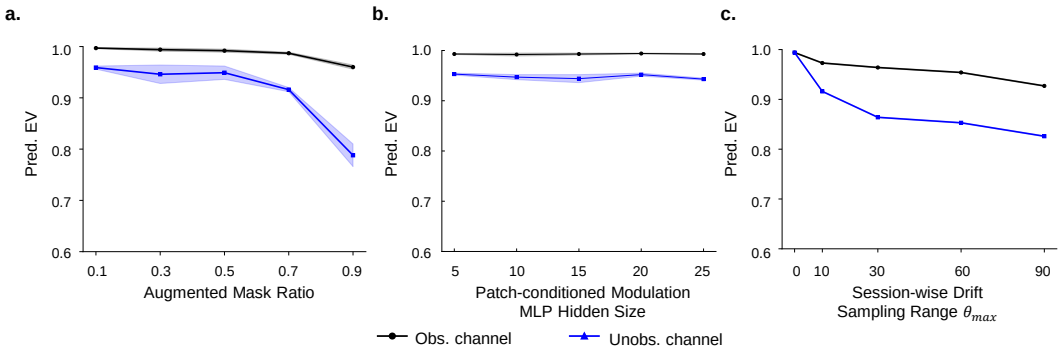

Figure 2: **Sensitivity analysis and robustness evaluation on Benchmark 2 (mask ratio = 0.5).** (a) Effect of the augmented masking ratio on prediction performance. Performance remains relatively stable under moderate masking ratios but degrades when excessive masking removes information needed to capture cross-channel dependencies. (b) Effect of the hidden size of the patch-conditioned modulation MLP. Performance is largely insensitive to the hidden size, suggesting that the conditioning module does not dominate the learned dynamics. (c) Robustness to cross-session channel misalignment, simulated by session-specific rotations with maximum angle ($\theta_{\max}$). Performance degrades gradually as the misalignment increases, while remaining relatively robust under substantial perturbations. Black and blue curves denote observed and unobserved channels, respectively.

Table 1: Ablation study for Benchmark 1 Exp 1.

Model Variant	**Obs. Pred. EV**	**Unobs. Pred. EV**
CHAMA	$\mathbf{0.9495 \pm 0.0118}$	$\mathbf{0.8536 \pm 0.0290}$
w/o temporal patching	0.8227 ± 0.0141	0.7303 ± 0.0336
w/o simulated missingness	$\underline{0.9461 \pm 0.0160}$	0.8122 ± 0.0651
w/o latent prediction	0.9374 ± 0.0132	$\underline{0.8520 \pm 0.0332}$
w/o available mask	0.9407 ± 0.0092	0.8202 ± 0.0238
w/o $\mathbf{p}_c$	0.9067 ± 0.0125	0.7292 ± 0.0231

importance of individual components varies with the redundancy structure of the data and the evaluation objective.

On Benchmark 1, where channels contain non-redundant information, CHAMA consistently achieves the best performance across both observed and unobserved channel prediction in both experiments. Removing temporal patching leads to substantial performance degradation. This highlights the importance of patch-level temporal modeling for capturing temporal dynamics when cross-channel redundancy is limited. Simulated missingness is similarly critical in this setting. In Exp.2, removing simulated missingness results in negative EV for both observed and unobserved channels, indicating a severe failure in prediction under incomplete observations. These results suggest that training with simulated missingness encourages robust inference from partially observed signals, especially when information cannot be readily recovered from other channels.

In contrast, Benchmark 2 contains redundant cross-channel information, making performance generally more robust to the removal of individual components. Most ablated variants retain relatively high prediction EV, suggesting that correlated observations can partially compensate for missing architectural components. Nevertheless, removing latent prediction or the available mask reduces unobserved prediction EV, indicating that explicit latent dynamics modeling and awareness of channel availability remain beneficial even when cross-channel redundancy is present.

The channel embedding p_c provides stable channel identities, allowing the model to distinguish shared mask tokens associated with different missing channels. Removing p_c substantially degrades prediction performance on Benchmarks 1 and 2. These results highlight the importance of explicit channel identity information for inferring unobserved channel dynamics, regardless of whether cross-channel information is redundant.

Table 2: Ablation study for Benchmark 1 Exp 2 (fully observed data ratio = 0.2).

Model Variant	Obs. Pred. EV	Unobs. Pred. EV
CHAMA	**0.8862 ± 0.0150**	**0.6795 ± 0.0440**
w/o temporal patching	0.6104 ± 0.0207	0.3374 ± 0.0517
w/o simulated missingness	-0.4959 ± 0.3180	-0.8038 ± 0.4312
w/o latent prediction	0.8625 ± 0.0292	0.5624 ± 0.0690
w/o available mask	0.8727 ± 0.0128	0.6147 ± 0.0509
w/o $\mathbf{p}_c$	0.7721 ± 0.0271	0.4329 ± 0.0739

Table 3: Ablation study for Benchmark 2 (mask ratio = 0.5).

Model Variant	Obs. Pred. EV	Unobs. Pred. EV
CHAMA	**0.9754 ± 0.0040**	**0.9349 ± 0.0091**
w/o temporal patching	0.9575 ± 0.0077	0.9082 ± 0.0145
w/o simulated missingness	0.9740 ± 0.0074	0.9160 ± 0.0202
w/o latent prediction	0.9677 ± 0.0054	0.8941 ± 0.0040
w/o patch-conditioned decoder	0.9676 ± 0.0066	0.9303 ± 0.0105
w/o available mask	0.9611 ± 0.0068	0.8889 ± 0.0089
w/o $\mathbf{p}_c$	0.8928 ± 0.0156	0.7489 ± 0.0364

For the real-world Benchmarks 3–5, the ablation results reveal a more nuanced relationship between reconstruction quality and downstream decoding performance. On Benchmark 3, removing temporal patching slightly improves unobserved reconstruction EV, while removing the available mask yields a marginally higher reconstructed signal decoding R^2. However, CHAMA achieves the highest latent state decoding R^2, suggesting that the full architecture better preserves information relevant to latent state decoding. On Benchmark 4, several ablated variants outperform the full model on individual metrics, particularly the variant without channel embeddings, which achieves the highest performance across all three metrics. On Benchmark 5, removing the available mask or channel embeddings improves unobserved reconstruction EV, whereas CHAMA achieves the highest reconstructed signal and latent state decoding R^2. These results suggest that reconstruction accuracy and downstream decoding performance do not always improve together, and that the benefits of individual components depend on the characteristics of the dataset and the evaluation objective.

Across the real-world benchmarks, removing simulated missingness consistently reduces unobserved reconstruction EV, with particularly pronounced degradation on Benchmarks 3 and 4. This further supports its role in improving robustness to incomplete observations. In contrast, the effects of temporal patching, latent prediction, the available mask, and channel embeddings are more dataset-dependent. In particular, although channel embeddings are important for prediction in Benchmarks 1 and 2, their removal can improve certain reconstruction and decoding metrics on real-world data, suggesting that explicit channel identity information is not equally beneficial across all settings.

Overall, the ablation study demonstrates that CHAMA's architectural components are particularly important in non-redundant settings, where robust inference from incomplete observations, effective temporal modeling, and explicit channel identity information are essential. In redundant settings, cross-channel correlations can mitigate the impact of removing certain components, although latent dynamics modeling and channel identity information remain beneficial. For real-world datasets, the results further highlight that reconstruction fidelity and downstream decoding performance capture complementary aspects of representation quality. While individual ablations can outperform the full model on specific metrics, CHAMA generally provides competitive performance across reconstruction and decoding objectives, with its full architecture being particularly beneficial for prediction under limited cross-channel redundancy.

F Implementation Details of Baselines

For fair comparison, the zero-padding and conditional adapter baselines share the same latent dynamical model as our method. The differences lie primarily in how observations are encoded

Table 4: Ablation study for Benchmark 3 (mask ratio = 0.2).

Model Variant	Unobs. Recon. EV	Recon. Signal Decoding R^2	Latent State Decoding R^2
CHAMA	0.6277 ± 0.0130	$\underline{0.5119 \pm 0.0113}$	$\mathbf{0.4626 \pm 0.0024}$
w/o temporal patching	$\mathbf{0.6317 \pm 0.0034}$	0.4983 ± 0.0077	0.4293 ± 0.0168
w/o simulated missingness	0.2646 ± 0.0252	0.4176 ± 0.0105	0.4121 ± 0.0165
w/o latent prediction	0.6176 ± 0.0127	0.5076 ± 0.0047	0.4426 ± 0.0100
w/o available mask	$\underline{0.6283 \pm 0.0064}$	$\mathbf{0.5126 \pm 0.0077}$	$\underline{0.4578 \pm 0.0163}$
w/o patch-conditioned decoder	0.6210 ± 0.0070	0.4931 ± 0.0111	0.4556 ± 0.0096
w/o $\mathbf{p}_c$	0.5999 ± 0.0092	0.4756 ± 0.0246	0.3999 ± 0.0407

Table 5: Ablation study for Benchmark 4 (mask ratio = 0.2).

Model Variant	Unobs. Recon. EV	Recon. Signal Decoding R^2	Latent State Decoding R^2
CHAMA	$\underline{0.3886 \pm 0.0126}$	$\underline{0.5464 \pm 0.0141}$	0.4399 ± 0.0082
w/o temporal patching	0.3735 ± 0.0201	0.5112 ± 0.0092	0.4112 ± 0.0140
w/o simulated missingness	0.2335 ± 0.0543	0.5072 ± 0.0163	$\underline{0.4596 \pm 0.0066}$
w/o latent prediction	0.3833 ± 0.0063	0.5432 ± 0.0186	0.4452 ± 0.0219
w/o available mask	0.3359 ± 0.0316	0.5331 ± 0.0218	0.4584 ± 0.0162
w/o $\mathbf{p}_c$	$\mathbf{0.4108 \pm 0.0241}$	$\mathbf{0.5546 \pm 0.0176}$	$\mathbf{0.4841 \pm 0.0120}$

into and decoded from the latent space, as well as how session variability is handled. In contrast, NeuroPaint follows a masked autoencoder paradigm and does not explicitly model latent dynamics.

F.1 Conditional adapter baseline

Without session variability (B1). For each missing-channel configuration, we learn a lightweight adapter that maps the observed subset of channels into a shared latent space. The latent dynamics and decoder are identical to those used in our method. A shared decoder is used across sessions, enabling reconstruction of missing channels from latent states. This variant does not explicitly model session-specific observation drift, and instead relies on the adapter to align different observation spaces.

With session variability (B2). As in B1, an adapter is learned for each observation configuration. In addition, each session is equipped with its own decoder (readout) that maps latent states to that session's observed channels. While this improves flexibility in modeling session-specific observation drift, it prevents reconstruction of unobserved channels, since the readout is session-specific and only defined for the channels observed in that session.

F.2 Zero-padding

Missing channels are replaced with zeros, and the resulting fixed-dimensional observations are passed through a shared encoder. A single decoder (readout) is used across all sessions to map latent states to the full observation space. Because the readout is shared, the model can in principle reconstruct unobserved channels from the inferred latent dynamics.

F.3 NeuroPaint baseline

We additionally compare against **NeuroPaint** [39], a masked autoencoder (MAE)-based method designed for neural signal inpainting. NeuroPaint randomly masks subsets of observed channels and trains an encoder–decoder architecture to reconstruct the missing signals from the visible ones. The method focuses on observation-level reconstruction and does not explicitly model latent temporal dynamics.

In the original formulation, NeuroPaint incorporates hierarchical structure (e.g., aggregating from single neurons to brain regions). In our setting, we adapt the method to a single-level representation, omitting hierarchical aggregation while preserving the core masked reconstruction objective. This comparison evaluates how accurately missing signals can be recovered, while underscoring the benefits of capturing latent dynamics rather than relying solely on observation-level reconstruction.

Table 6: Ablation study for Benchmark 5 (mask ratio = 0.2).

Model Variant	Unobs. Recon. EV	Recon. Signal Decoding R^2	Latent State Decoding R^2
CHAMA	0.6855 ± 0.0139	$\mathbf{0.4964 \pm 0.0183}$	$\mathbf{0.4844 \pm 0.0854}$
w/o temporal patching	0.6846 ± 0.0177	$\underline{0.4937 \pm 0.0211}$	0.4729 ± 0.0283
w/o simulated missingness	0.6778 ± 0.0274	0.3803 ± 0.0476	0.3547 ± 0.0236
w/o latent prediction	0.6836 ± 0.0142	0.4925 ± 0.0223	0.4736 ± 0.0194
w/o available mask	$\underline{0.6913 \pm 0.0159}$	0.4550 ± 0.0437	0.4495 ± 0.0599
w/o $\mathbf{p}_c$	$\mathbf{0.6985 \pm 0.0306}$	0.4347 ± 0.0997	$\underline{0.4739 \pm 0.0403}$

G Implementation Details of Latent Representation Evaluation

To quantitatively evaluate the quality of inferred latent representations, we employ a linear readout model. Given inferred latent variables $\mathbf{z}_t \in \mathbb{R}^d$ and target variables $\mathbf{y}_t \in \mathbb{R}^k$, we fit a linear mapping:

$$\hat{\mathbf{y}}_t = \mathbf{W}\mathbf{z}_t + \mathbf{b}, \tag{53}$$

where $\mathbf{W} \in \mathbb{R}^{k \times d}$ and $\mathbf{b} \in \mathbb{R}^k$ are learned parameters.

We estimate $\mathbf{W}$ and $\mathbf{b}$ by minimizing the mean squared error (MSE):

$$\min_{\mathbf{W},\mathbf{b}} \frac{1}{T} \sum_{t=1}^{T} \|\mathbf{y}_t - (\mathbf{W}\mathbf{z}_t + \mathbf{b})\|_2^2 . \tag{54}$$

The solution is obtained via ordinary least squares.

Evaluation Metric. We report the coefficient of determination (R^2):

$$R^2 = 1 - \frac{\sum_t \|\mathbf{y}_t - \hat{\mathbf{y}}_t\|_2^2}{\sum_t \|\mathbf{y}_t - \bar{\mathbf{y}}\|_2^2}, \tag{55}$$

where $\bar{\mathbf{y}}$ denotes the empirical mean of the target variable.

Evaluation Protocol. The reported R^2 reflects in-sample goodness of fit and is intended as a diagnostic measure of how well the inferred representations linearly capture the target variables, rather than a measure of generalization performance.

- **B2 (known latent dynamics):** The data are generated from a 3-dimensional Lorenz system, whose latent states are linearly projected into a 20-dimensional observation space to produce the observed signals. The targets $\mathbf{y}_t$ correspond to the underlying 3D latent states. We evaluate how well the inferred latents recover these dynamics using a linear readout and report the R^2.
- **B3, B4 and B5 (unknown latent dynamics):** The targets $\mathbf{y}_t$ correspond to a behavioral variable, namely the 2D velocity from the datasets. We train separate linear readouts from (i) the inferred latent space and (ii) the reconstructed observation space, and report decoding performance in terms of R^2.

H Implementation Details of CHAMA

All experiments were implemented in PyTorch and conducted on a workstation equipped with NVIDIA RTX 4090 GPUs. The network architecture and training settings across all benchmarks are summarized in Table 7. Benchmark-specific hyperparameters are summarized in Table 8.

Table 7: Architectural and training settings shared across benchmarks.

Architectural settings		Training settings	
Encoder MLP activation function	ReLU	Optimizer	Adam
Encoder MLP layer number	2	Learning rate	1e-3
Dynamics MLP activation function	Tanh	Batch size	50
Dynamics MLP layer number	3	Mask augmentation ratio	0.3–0.7
Dropout	0.0	Early stopping patience	50
Dynamics order	1	Train/Validation/Test split	60/20/20

Table 8: Benchmark-specific hyperparameters.

Hyperparameter	**Bench. 1**	**Bench. 2**	**Bench. 3**	**Bench. 4**	**Bench. 5**
Attention embedding dimension (d_a)	10	20	4	4	4
Latent dimension (d_z)	32	32	48	64	32
MLP hidden dimension (d_h)	32	32	256	256	128
Patch length (L)	10	10	5	5	5
Teacher forcing step (K_{tf})	250	250	5	1	1
Epochs	1000	1000	200	200	200
Loss function	MSE	MSE	L1	L1	L1
Decoder LoRA rank	–	8	8	–	–
Behavior delay	–	–	5	5	5
$\mathcal{L}_{\text{lat}}$ weight	1	1	0.01	0.01	0.01
$\mathcal{L}_{\text{sig-pred}}$ weight	1	1	1	1	1
$\mathcal{L}_{\text{rec}}$ weight	1	1	0.01	0.01	0.01

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes. The abstract and introduction accurately summarize the paper's scope and contributions: CHAMA is presented as a mask-conditioned framework for heterogeneous channel availability, using a single global observation interface, temporal aggregation, and bidirectional coupling between missing channels and latent dynamics. These claims match the method sections and are supported by experiments on synthetic Lorenz systems and real multi-session neural recordings, covering missing-channel reconstruction, forecasting, latent fidelity, and downstream decoding. The paper also states limitations in the conclusion, such as assuming shared latent dynamics and aligned channel identities, so the scope is not overstated.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes. The paper explicitly discusses limitations in the Discussion and Conclusion section.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper includes a dedicated theoretical section in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes. We provide implementation details in Sec. 4 and Appendix C, F, G, H. And we upload the codes and model to cover the results. All datasets are publicly accessible.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper provides an anonymized source-code repository link in the abstract. The synthetic datasets are generated from fully specified Lorenz-system procedures, and the real dataset is publicly released and cited.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Sec. 4 and Appendix C provide dataset-generation and preprocessing details, while Appendices F, G and H provides implementation details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports variability information in several experimental figures using shaded regions corresponding to standard deviation. The experimental figures throughout the paper consistently visualize performance trends together with uncertainty estimates, which helps assess the robustness of the reported results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All training, testing, and inference experiments were conducted on a server equipped with four NVIDIA RTX 4090 GPUs.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses synthetic data and previously released neural recording data, and is intended to improve robustness and interpretability of partially observed neural dynamics models. The authors have reviewed the NeurIPS Code of Ethics and are not aware of any deviations.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper focuses on academic and uses public datasets. We do not foresee any potential negative societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release high-risk pretrained generative models, scraped datasets, or dual-use foundation models. The released code is for reproducing experiments on synthetic and previously released neural recording data.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper cites the existing datasets and baselines used in the experiments, including the center-out reaching dataset and NeuroPaint baseline.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The paper introduces and releases code for CHAMA through an anonymized repository. The method, training objective, datasets, baselines, and evaluation protocols are documented in the manuscript and appendices.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not conduct crowdsourcing or collect new human-subject data. The experiments use previously collected neural recording datasets, including two non-human primate datasets (B3 and B4) and one human iBCI dataset (B5). No new participant recruitment or data collection was conducted as part of this study.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This study does not involve new experiments with human participants or new data collection. All analyses are conducted using previously collected datasets, including two non-human primate neural recording datasets (B3 and B4) and a previously published human iBCI dataset (B5). Ethical approvals and informed consent for the original human data collection should be documented by the original dataset publication.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: In this study, LLMs were employed to provide non-scientific writing support, including grammar verification and language refinement, primarily aimed at enhancing textual readability. The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.