Integrating Multimodal Data for Joint Generative Modeling of Complex Dynamics

Manuel Brenner [1 2] Florian Hess [1 2] Georgia Koppe [* 3 4] Daniel Durstewitz [* 1 2 3]

Abstract

Many, if not most, systems of interest in science are naturally described as nonlinear dynamical systems. Empirically, we commonly access these systems through time series measurements. Often such time series may consist of discrete random variables rather than continuous measurements, or may be composed of measurements from multiple data modalities observed simultaneously. For instance, in neuroscience we may have behavioral labels in addition to spike counts and continuous physiological recordings. While by now there is a burgeoning literature on deep learning for dynamical systems reconstruction (DSR), multimodal data integration has hardly been considered in this context. Here we provide such an efficient and flexible algorithmic framework that rests on a multimodal variational autoencoder for generating a sparse teacher signal that guides training of a reconstruction model, exploiting recent advances in DSR training techniques. It enables to combine various sources of information for optimal reconstruction, even allows for reconstruction from symbolic data (class labels) alone, and connects different types of observations within a common latent dynamics space. In contrast to previous multimodal data integration techniques for scientific applications, our framework is fully *generative*, producing, after training, trajectories with the same geometrical and temporal structure as those of the ground truth system.

---

[*]Equal contribution [1]Dept. of Theoretical Neuroscience, Central Institute of Mental Health (CIMH), Medical Faculty Mannheim, Heidelberg University, Germany [2]Faculty of Physics and Astronomy, Heidelberg University, Germany [3]Interdisciplinary Center for Scientific Computing, Heidelberg University, Germany [4]Hector Institute for AI in Psychiatry, CIMH. Correspondence to: Manuel Brenner <manuel.brenner@zi-mannheim.de>, Daniel Durstewitz <daniel.durstewitz@zi-mannheim.de>.

1. Introduction

For many temporally evolving complex systems in physics, biology, or the social sciences, we have only limited knowledge about the generating dynamical mechanisms. Inferring these from data is a core interest in any scientific discipline. It is also practically highly relevant for predicting important changes in system dynamics, like tipping points in climate systems (Bury et al., 2021; Patel & Ott, 2023). In recent years, a variety of machine learning (ML) methods for recovering dynamical systems (DS) directly and automatically from time series observations have been proposed (Brunton et al., 2016; Vlachas et al., 2018; Lusch et al., 2018; Pathak et al., 2018; Koppe et al., 2019b; Schmidt et al., 2021; Fu et al., 2019; Vlachas et al., 2020; Gauthier et al., 2021; Jordana et al., 2021; Lejarza & Baldea, 2022; Brenner et al., 2022; Hess et al., 2023; Chen & Wu, 2023; Yang et al., 2023; Vlachas et al., 2022), mostly based on recurrent neural networks (RNNs) for approximating the flow of the underlying true DS (Vlachas et al., 2020; Koppe et al., 2019b; Schmidt et al., 2021; Brenner et al., 2022; Hess et al., 2023; Vlachas et al., 2022). However, almost all of these methods assume that observed time series come as continuous signals with Gaussian noise from a single type of source. Yet, time series data from discrete random processes, like class labels or counts, are in fact quite commonplace in many areas, e.g., in the medical domain (electronic health records, smartphone-based data) (Koppe et al., 2019a), neuroscience (behavioral responses) (Schneider et al., 2023), or climate science (event counts) (Tziperman et al., 1997). Moreover, with the increasing availability of massive data acquisition techniques in many scientific disciplines, often one has simultaneous measurements from multiple different types of data channels with different statistical properties. For instance, in neuroscience one may have simultaneous recordings of spike events and calcium imaging together with behavioral class labels.

Multimodal data integration In general, multimodal data integration is a thriving topic in many areas of artificial intelligence and ML research (Ahuja et al., 2017; Bhagwat et al., 2018; Baltrusaitis et al., 2019; Sutter et al., 2020; Shi et al., 2020; Antelmi et al., 2018; Liang et al., 2022; Lipkova et al., 2022; Xu et al., 2023; Steyaert et al., 2023; Warner

et al., 2023; OpenAI, 2023). Generative models fusing multiple data channels produce a common latent code which allows for cross-modal prediction or output generation (e.g., images from language; Amrani et al. (2021); Radford et al. (2021)), improves inference by complementing information too noisy or missing in one channel through recordings from other modalities (Qian et al., 2022), or may reveal interesting links among observed modalities (Liang et al., 2015).

Variational autoencoders (VAE) (Kingma & Welling, 2014; Rezende et al., 2014) are one popular variant of generative models which naturally lend themselves to multimodal settings (Baltrusaitis et al., 2019; Wu & Goodman, 2018; Sutter et al., 2020) and a sequential formulation (Bayer et al., 2021; Girin et al., 2021; Bai et al., 2021). Longitudinal autoencoders have been proposed (Ramchandran et al., 2021) to model temporal correlations in latent space, and have also been extended to multimodal data (Öğretir et al., 2022) for the purpose of time series forecasting (Antelmi et al., 2018; Bhagwat et al., 2018; Dezfouli et al., 2018; Shi et al., 2020; Sutter et al., 2020; Qian et al., 2022).

Dynamical Systems Reconstruction (DSR) DSR, however, goes beyond mere forecasting in that we request a dynamical model of the data-generating process which captures the temporal and geometrical structure of the true underlying system (Koppe et al., 2019b; Abarbanel, 2022; Platt et al., 2023). A successful DS reconstruction can reveal important invariant topological and geometrical characteristics of the state space the DS lives in, its long-term temporal behavior (attractors), and its sensitivity to perturbations and parameter variations. It provides a mechanistic surrogate for the observed system which can be further analyzed, simulated, perturbed, or lesioned to obtain additional insight into its inner workings (Durstewitz et al., 2023).

Often special training techniques (Mikhaeil et al., 2022; Brenner et al., 2022; Hess et al., 2023; Abarbanel, 2013; 2022; Vlachas & Koumoutsakos, 2023), regularization approaches (Schmidt et al., 2021; Platt et al., 2022; Jiang et al., 2023; Platt et al., 2023), or structural priors as in physics-informed neural networks (PINNs; Raissi et al. (2019)), are required to achieve these goals. Plain 'out-of-the-box' gradient descent techniques or standard variational inference will often fail to reconstruct the underlying system's long-term properties (Arribas et al., 2020; Brenner et al., 2022; Hess et al., 2023; Platt et al., 2023; Jiang et al., 2023; Durstewitz et al., 2023), as these standard procedures do not sufficiently constrain the vector field to prevent true and reconstructed trajectories from quickly diverging. This is especially a problem in chaotic systems, where it is related to the exploding or vanishing gradient problem in RNN training (Hochreiter & Schmidhuber, 1997; Hochreiter et al., 2001; Bengio et al., 1994; Mikhaeil et al., 2022;

Hess et al., 2023). Control-theoretic techniques are often used to keep model-generated trajectories on track, or to synchronize them with those produced by the observed system, whilst training (Voss et al., 2004; Abarbanel et al., 2009; Abarbanel, 2013; Verzelli et al., 2021; Platt et al., 2021; Abarbanel, 2022). Similar ideas underlie the recently proposed methods of sparse (Mikhaeil et al., 2022; Brenner et al., 2022) and generalized (Hess et al., 2023) teacher forcing (STF and GTF, respectively) for learning chaotic DS, which balance model-generated and data-inferred states during training in a way that optimally controls gradient flows and trajectory divergence (Mikhaeil et al., 2022; Hess et al., 2023; Eisenmann et al., 2023).

Specific contributions Although important in many areas of science, an efficient framework for multimodal data integration *for the purpose of DS reconstruction* is currently essentially lacking. A particular challenge here is to reconcile highly efficient training techniques for DS reconstruction like STF or GTF, which usually rest on an invertible decoder model, with the need to infer a common latent code from many different types of data. Here we address this challenge through a novel formulation of the multimodal data integration problem for DS reconstruction: Building on the success of VAEs for multimodal integration (Baltrusaitis et al., 2019; Wu & Goodman, 2018; Sutter et al., 2020), we use multimodal VAEs (MVAEs) to construct a common latent representation from random variables following different distributional models. Crucially, however, rather than constructing a sequential VAE (SVAE) process that directly operates on this latent code (Kramer et al., 2022), here we employ the MVAE *to create a multimodal TF signal* for guiding a DSR model in training. Special loss terms and shared decoder models ensure consistency between the MVAE's and reconstruction model's latent codes. This not only efficiently exploits the information available in multiple different data streams for DS reconstruction (as compared to a variety of alternative techniques), but for the first time, to our knowledge, enables DS reconstructions of chaotic systems from categorical information alone.

2. Multimodal Teacher Forcing (MTF)

Techniques like STF (Brenner et al., 2022; Mikhaeil et al., 2022) or GTF (Hess et al., 2023) enable reconstructions even from challenging real-world data on which many previous methods failed. Hence, the key idea for making DS reconstruction from multiple, statistically distinct data sources work, is the generation of a *multimodal TF signal* for efficient training of a reconstruction model. For this we utilize an MVAE, which is then trained jointly with the DSR model through a set of shared decoder models. This general framework, illustrated in Fig. 1, is very flexible and modular. In principle it can be used with any type of DSR model for

approximating the flow of the observed system, as well as with any set of decoder and encoder models.

DSR model   While completely flexible, for our specific experiments we chose the recently introduced 'dendritic piecewise linear RNN (dendPLRNN)' (Brenner et al., 2022) as a reconstruction model, as it is both expressive and has a mathematically tractable, piecewise linear design (Monfared & Durstewitz, 2020; Eisenmann et al., 2023) (other types of RNNs, such as LSTMs or GRUs, may be used and give similar results, see Appx. Table B3). The dendPLRNN is defined by the M-dimensional latent process equation

$$
\begin{aligned}
\boldsymbol{z}_t =& \boldsymbol{A}\boldsymbol{z}_{t-1} + \boldsymbol{W}\sum_{b=1}^{B}\alpha_b\max(\boldsymbol{0}, \boldsymbol{z}_{t-1} - \boldsymbol{h}_b) \\
& + \boldsymbol{h} + \boldsymbol{U}\boldsymbol{s}_t + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}),
\end{aligned}
\tag{1}
$$

which describes the temporal evolution of an M-dimensional latent state vector $\boldsymbol{z}_t = (z_{1t} \ldots z_{Mt})^T$ with diagonal matrix $\boldsymbol{A} \in \mathbb{R}^{M \times M}$, off-diagonal matrix $\boldsymbol{W} \in \mathbb{R}^{M \times M}$, noise covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{M \times M}$, a nonlinearity given by the 'dendritic' spline expansion with slopes $\alpha_b \in \mathbb{R}$ and thresholds $\boldsymbol{h}_b \in \mathbb{R}^M$ (Brenner et al., 2022), and a linear term $\boldsymbol{U} \in \mathbb{R}^{M \times P}$ weighing P-dimensional external inputs $\boldsymbol{s}_t$.

Decoder models   To infer this latent process equation jointly from multiple data modalities, the dendPLRNN is connected to different decoder models that take the distinct distributional properties of each modality into account. Our formulation is general, i.e. can work with any combination of continuous and/or discrete data types, and the decoders may take the form of any differentiable parametric probability model $p_\theta(\bullet | \boldsymbol{Z})$. For our exposition, however, we consider as concrete example time series of multivariate Gaussian ($\boldsymbol{X}$), ordinal ($\boldsymbol{O}$),[1] and count ($\boldsymbol{C}$) nature of length T, $\boldsymbol{Y} = \{\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}; \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}; \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_T\}\}$, where modalities $\boldsymbol{x}_t \in \mathbb{R}^{N_1}$, $\boldsymbol{o}_t \in \mathbb{Z}^{N_2}$, and $\boldsymbol{c}_t \in \mathbb{N}^{N_3}$ may all have different dimensions. In this case, we may take as decoders a linear Gaussian model, a cumulative link model for ordinal data, and a log-link function for Poisson data:

$$
\boldsymbol{x}_t | \boldsymbol{z}_t \sim \mathcal{N}(\boldsymbol{B}\boldsymbol{z}_t, \boldsymbol{\Gamma}) ;
\tag{2}
$$

$$
\boldsymbol{o}_t | \boldsymbol{z}_t \sim \mathrm{Ordinal}(\boldsymbol{\beta}\boldsymbol{z}_t, \epsilon);
\tag{3}
$$

$$
\boldsymbol{c}_t | \boldsymbol{z}_t \sim \mathrm{Poisson}(\lambda(\boldsymbol{z}_t)),
\tag{4}
$$

The DSR model loss is now given by the sum of the negative log-likelihoods of the decoders, assuming conditional

---

[1]Ordinal data, as often encountered in psychology or economics (Likert, 1932), are non-metric but still ordered.

independence given latent states $\boldsymbol{z}$:

$$
\begin{aligned}
\mathcal{L}_{\mathrm{DSR}} = -\sum_{t=1}^{T}(&\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_t | \boldsymbol{z}_{1:K,t}) + \\
& \log p_{\boldsymbol{\theta}}(\boldsymbol{o}_t | \boldsymbol{z}_{1:K,t}) + \log p_{\boldsymbol{\theta}}(\boldsymbol{c}_t | \boldsymbol{z}_{1:K,t})),
\end{aligned}
\tag{5}
$$

where only the first K latent states are used to allow for sharing with the MVAE as explained below, and the individual likelihood terms are specified in Appx. A.1.

Training: Multimodal Teacher Forcing (MTF)   Training RNNs on time series is generally challenging due to the exploding and vanishing gradient problem (Bengio et al., 1994). This becomes particularly severe in DS reconstruction as for chaotic systems exploding gradients cannot be avoided even in principle, due to the positive maximum Lyapunov exponent resulting in exponential divergence of trajectories (Mikhaeil et al., 2022; Hess et al., 2023). Yet, at the same time batch length cannot be too short, as the training algorithm will then fail to capture the system's long term behavior and limit sets (Brenner et al., 2022; Platt et al., 2022; 2023). STF (Mikhaeil et al., 2022) and GTF (Hess et al., 2023) therefore balance loss and trajectory divergence with the need to capture relevant long time scales in an optimal way. They do so by replacing forwarded-iterated latent states $\boldsymbol{z}_t$ by data-inferred states $\hat{\boldsymbol{z}}_t$ sparsely, that is only at strategically chosen time points determined from the empirically estimated maximum Lyapunov exponent in the case of STF (Mikhaeil et al., 2022), or by averaging forward-propagated and data-inferred states in an optimal way in the case of GTF (Hess et al., 2023). Data-inferred states are obtained by inverting the decoder model (e.g., $\hat{\boldsymbol{z}}_t = (\boldsymbol{B}^T\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{x}_t$ for a simple linear-Gaussian model as in Eq. 2, i.e. taking the pseudo-inverse). This forcing is then only applied during model training, not at run time (see Methods A.1 for further details).

Inverting the decoder model is in general not possible, however, in particular for discrete random variables. Moreover, in the case of multiple simultaneously observed data modalities, it is unclear how to combine the different data modalities to obtain an optimal estimate $\hat{\boldsymbol{z}}_t$. We therefore need another means to create a (sparse) TF signal if we would like to utilize STF or GTF for training. Here we achieve this through an MVAE for building a joint latent representation over the different data types, $\tilde{\boldsymbol{z}}_t \sim p(\tilde{\boldsymbol{z}}_t | \boldsymbol{X}, \boldsymbol{O}, \boldsymbol{C})$, which can be used as a TF signal at time t. We denote the MVAE states by $\tilde{\boldsymbol{z}}_t \in \mathbb{R}^K$ to avoid confusion with the latent dynamical process $\boldsymbol{z}_t \in \mathbb{R}^M$ generated by the reconstruction model.[2] To ensure consistency between the DSR model's

---

[2]Note that in general we do not assume the MVAE and reconstruction model latent codes to have the same dimensionality, i.e. $K \leq M$, which, e.g., could enable the DSR model to capture additional, unobserved latent states or non-stationarity in the data; see Methods A.1 for further details.
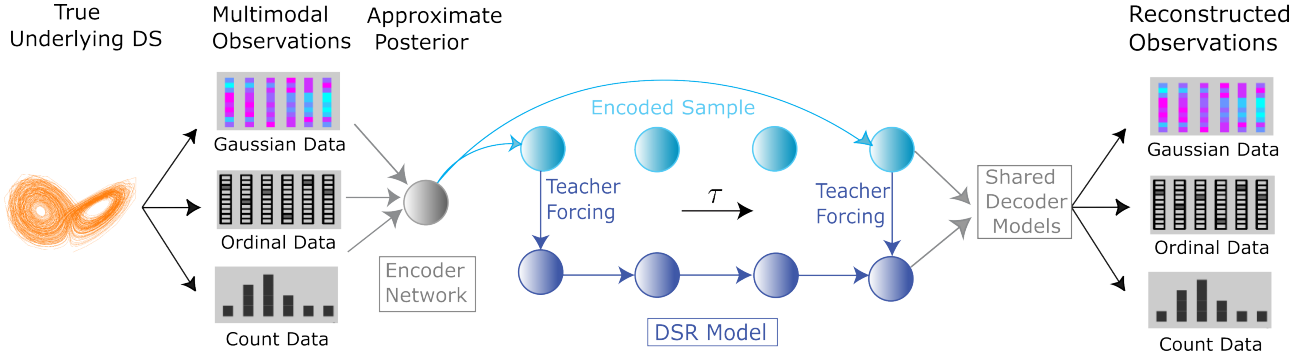
Figure 1. MTF setup. Multimodal observations are translated via an encoder into a common latent representation, which is used for sparse TF in the DSR model's latent space. The latent trajectory is then mapped back into observation space via modality-specific decoder models, which are shared between the MVAE and DSR model.

and the MVAE's latent states, the MVAE is coupled to the observations through the *very same* set of decoder models

$$x_t|\tilde{z}_t \sim \mathcal{N}(B\tilde{z}_t, \Gamma); \tag{6}$$
$$o_t|\tilde{z}_t \sim \text{Ordinal}(\beta\tilde{z}_t, \epsilon); \tag{7}$$
$$c_t|\tilde{z}_t \sim \text{Poisson}(\lambda(\tilde{z}_t)), \tag{8}$$

sharing all decoder parameters (i.e. B, Γ, β etc.) with the DSR model (see Fig. 1).

The MVAE is trained by minimizing the negative Evidence Lower Bound (ELBO)

$$\mathcal{L}(\phi, \theta; Y) = -\mathbb{E}_{q_\phi}[\log p_\theta(Y|\tilde{Z}) + \log p_\theta(\tilde{Z})] - \mathbb{H}_{q_\phi}(\tilde{Z}|Y) \tag{9}$$

using the reparameterization trick for latent random variables (Kingma & Welling, 2014), where $\mathbb{H}_{q_\phi}$ is the entropy of the approximate posterior, and assuming conditional independence of the observations given the latent states. We tested various choices for the encoder model $q_\phi(\tilde{Z}|Y)$, including mixture-of-experts, RNNs or transformers, but achieved best results with a temporal CNN with all data modalities concatenated (see Table B2 for results and Methods A.1 for architectural details).

Process prior and total loss We thus have specified $p_\phi(Y|\tilde{Z})$ through our choice of decoder models, and parameterized $q_\phi(\tilde{Z}|Y)$ by a temporal CNN. But what is the best choice for the prior $p_\theta(\tilde{Z})$? Crucially, to further enforce consistency between the MVAE and DSR model latent codes, $\tilde{Z}$ and Z, respectively, we assume that $p_\theta(\tilde{Z})$ is given through the DSR model (the dendPLRNN in our case). Specifically, in line with the Gaussian assumptions in the dendPLRNN, Eq. 1, we assume for the second term in

Eq. 9:

$$-\mathbb{E}_{q_\phi}[\log p_\theta(\tilde{Z})] \approx \frac{1}{L}\sum_{l=1}^{L}\sum_{t=1}^{T}\frac{1}{2}\Big(\log|\Sigma| \\ + (\tilde{z}_t^{(l)} - \mu_t)^\top \Sigma^{-1}(\tilde{z}_t^{(l)} - \mu_t) \\ + const.\Big), \tag{10}$$

where the expectation value is approximated by L Monte Carlo samples $\tilde{z}_t^{(l)} \sim q_\phi(\tilde{z}_t|Y)$, and the means $\mu_t = \mathbb{E}(z_t|z_{t-1})$ directly come from the DSR model (see Methods A.1 for details). We therefore also call this term, Eq. 10, the *consistency loss*, $\mathcal{L}_{\text{con}}$.

The total training loss is now given by the MVAE's negative ELBO, Eq. 9, combined with the reconstruction loss (negative log-likelihood) of the DSR model (funneled through the same set of decoder models):

$$\mathcal{L}_{\text{MTF}} = -\mathbb{E}_{q_\phi}[\log p_\theta(Y|\tilde{Z})] - \mathbb{H}_{q_\phi}(\tilde{Z}|Y) \\ \underbrace{-\mathbb{E}_{q_\phi}[\log p_\theta(\tilde{Z})]}_{\mathcal{L}_{\text{con}}} \underbrace{-\log p_\theta(Y|Z)}_{\mathcal{L}_{\text{DSR}}} \tag{11}$$

Whilst training, we then use $\tilde{z}_t$ as our TF signal for guiding the DSR model. Specifically, for STF we replace (a subset of the) latent states z_t by $\tilde{z}_t$ at strategically chosen times $l\tau + 1$, $l \in \mathbb{N}_0$, as illustrated in Fig. 1 (similarly, for GTF we may take a weighted average; see Hess et al. (2023) for details).

Note that our usage of an MVAE here thus profoundly differs from the more conventional use in, for instance, SVAEs (Sutter et al., 2020; Kramer et al., 2022): The MVAE's major purpose is to generate an appropriate TF control signal which *guides* the DSR model in training, not in modeling the process itself as in an SVAE; see Fig. A6 for an illustration of this important difference. This control of model-generated trajectories through data-inferred values has been

Table 1. Comparison of dendPLRNN trained by MTF (proposed method), by a sequential multimodal VAE (SVAE) based on (Kramer et al., 2022), a VAE-TF approach similar to MTF except that all data modalities were 'Gaussianized' (GVAE-TF), BPTT-TF as in (Brenner et al., 2022) using Gaussianized data, and a multiple-shooting (MS) approach. Training was performed on multivariate normal, ordinal, and count data produced by the chaotic Lorenz system, Rössler system, and Lewis-Glass model. Values are mean $\pm$ SEM, averaged over 15 trained models. X = value cannot be computed for this model (e.g., because resp. decoder model is not present). Note that SCC (Spearman cross-correlation), OACF (ordinal autocorrelation function), and CACF (count autocorrelation function) all refer to mean-squared-errors (MSEs) between ground truth and generated correlation functions. Bold numbers indicate top performance within ±1 SEM.

Dataset	Method	$D_{stsp}\downarrow$	$D_H\downarrow$	PE$\downarrow$	OPE$\downarrow$	SCC$\downarrow$	OACF$\downarrow$	CACF$\downarrow$
Lorenz	MTF	**3.4 $\pm$ 0.35**	**0.30 $\pm$ 0.06**	**1.3e−2 $\pm$ 2e−4**	**0.12 $\pm$ 0.03**	**0.07 $\pm$ 0.01**	**0.07 $\pm$ 0.01**	**6.6e−5 $\pm$ 8.1e−6**
	SVAE	11.1 $\pm$ 0.6	0.82 $\pm$ 0.05	6.3e−1 $\pm$ 5.1e−2	0.68 $\pm$ 0.03	0.14 $\pm$ 0.01	0.18 $\pm$ 0.02	8.5e−5 $\pm$ 1.6e−5
	BPTT	6.31 $\pm$ 1.2	0.47 $\pm$ 0.11	2.1e−1 $\pm$ 2.4e−2	0.33 $\pm$ 0.04	0.11 $\pm$ 0.02	0.09 $\pm$ 0.02	8.2e−5 $\pm$ 9e−6
	MS	4.5 $\pm$ 1.5	0.61 $\pm$ 0.08	X	X	0.14 $\pm$ 0.04	0.11 $\pm$ 0.02	**6.5e−5 $\pm$ 3.8e−6**
	GVAE-TF	4.3 $\pm$ 0.3	0.47 $\pm$ 0.07	3.6e−1 $\pm$ 1.5e−3	X	X	X	X
	BPTT-TF	8.8 $\pm$ 1.9	0.86 $\pm$ 0.05	4.4e−1 $\pm$ 2.2e−2	X	X	X	X
Rössler	MTF	**1.45 $\pm$ 0.71**	**0.32 $\pm$ 0.03**	**1.9e−3 $\pm$ 7.1e−5**	**0.08 $\pm$ 0.02**	**0.04 $\pm$ 0.004**	**0.017 $\pm$ 0.003**	**6.5e−5 $\pm$ 1.2e−5**
	SVAE	10.7 $\pm$ 1.5	0.66 $\pm$ 0.05	1.5e−1 $\pm$ 3.1e−2	0.24 $\pm$ 0.02	0.17 $\pm$ 0.03	0.13 $\pm$ 0.02	1.1e−4 $\pm$ 1.4e−5
	BPTT	9.05 $\pm$ 0.5	0.7 $\pm$ 0.01	7.4e−2 $\pm$ 2.0e−3	0.18 $\pm$ 0.04	0.3 $\pm$ 0.03	0.19 $\pm$ 0.07	1.5e−4 $\pm$ 6e−6
	MS	3.99 $\pm$ 1.1	0.59 $\pm$ 0.04	X	X	0.08 $\pm$ 0.04	0.09 $\pm$ 0.02	1.6e−4 $\pm$ 5.9e−5
	GVAE-TF	12.1 $\pm$ 0.5	0.55 $\pm$ 0.04	4.9e−2 $\pm$ 3.4e−3	X	X	X	X
	BPTT-TF	8.9 $\pm$ 1.4	0.64 $\pm$ 0.07	2.8e−1 $\pm$ 1.8e−3	X	X	X	X
Lewis-Glass	MTF	0.27 $\pm$ 0.07	0.33 $\pm$ 0.02	**2.1e−3 $\pm$ 7e−5**	**0.11 $\pm$ 0.01**	0.12 $\pm$ 0.03	0.05 $\pm$ 0.02	2.3e−4 $\pm$ 2.0e−5
	SVAE	2.6 $\pm$ 0.5	0.52 $\pm$ 0.03	8.0e−2 $\pm$ 4e−3	0.26 $\pm$ 0.01	0.4 $\pm$ 0.05	0.18 $\pm$ 0.03	7.5e−3 $\pm$ 4.7e−3
	BPTT	2.8 $\pm$ 0.5	0.57 $\pm$ 0.05	6.2e−2 $\pm$ 3e−3	0.23 $\pm$ 0.02	0.47 $\pm$ 0.08	0.21 $\pm$ 0.03	9.1e−3 $\pm$ 3.2e−3
	MS	**0.33 $\pm$ 0.06**	**0.35 $\pm$ 0.01**	X	X	**0.08 $\pm$ 0.01**	**0.04 $\pm$ 0.002**	**1.9e−4 $\pm$ 7.5e−6**
	GVAE-TF	**0.28 $\pm$ 0.08**	0.44 $\pm$ 0.02	4.6e−3 $\pm$ 4e−4	X	X	X	X
	BPTT-TF	2.51 $\pm$ 0.71	0.43 $\pm$ 0.03	2.6e−2 $\pm$ 3e−3	X	X	X	X

found crucial for successful DSR (Mikhaeil et al., 2022; Brenner et al., 2022; Hess et al., 2023). This in turn necessitates the reconstruction loss $\mathcal{L}_{DSR}$, which allows for the DSR model to also make direct contact with the data, rather than only indirectly through the posterior approximation. Without it, much of the burden for recovering the dynamics would be on the (fallible) MVAE, while with it, through the sharing of decoder models, the approximate posterior will be improved through the DSR process. Ablation studies indeed further confirm that all components of the loss (Fig. A4), including $\mathcal{L}_{DSR}$, as well as an optimal choice for the TF interval (Fig. A7), are crucial to the success of the approach. Specifically, optimal TF intervals were associated with smooth and well navigable loss landscapes (Fig. A5).

3. Results

3.1. Evaluating MTF: Comparison to Other Algorithmic Strategies

We first compared MTF on moderately challenging multimodal DS reconstruction tasks[3] to a variety of other possible algorithmic strategies that may be considered, all of which lack one or more key ingredients of our methodology. For this, a training and a test set of $100,000$ time steps were created from a Lorenz-63 and a Rössler system with 1% process noise, and a 6d Lewis-Glass network model (Lewis & Glass, 1992; Gilpin, 2022), all in their chaotic regimes (see Methods A.5 for detailed parameter settings and numerical integration). From the simulated trajectories, we

[3]Moderately challenging because many of the comparison methods actually broke down on the more difficult setups.

then sampled ordinal and count observations using Eqs. 14 and 17 (with randomly drawn parameters), as well as continuous observations with 10% Gaussian noise. Example reconstructions by MTF are in Fig. 2**a** and Fig. A10.

We compared the performance of MTF on these simulated data to the following setups: First, to a multimodal SVAE (Kramer et al. (2022); to our knowledge the only other general approach intended for DSR from multimodal data). Second, we tried classical RNN training, where observations are provided as inputs at every time step, via truncated BPTT using modality-specific decoder models, Eqs. 2-4. A third approach that can deal with multimodal observation models but, unlike TF, does not require model inversion, is 'multiple shooting (MS)', a method suggested in the dynamical systems literature (Voss et al., 2004), but to our knowledge not extended to multimodal data so far. We also included two other comparisons where we first transformed multimodal data to be approximately Gaussian (via Box-Cox & Gaussian kernel smoothing, see Methods A.3), and then either trained the RNN via standard BPTT-TF (Brenner et al., 2022) or by VAE-TF, similar as in MTF, but without modality-specific decoder models (labeled GVAE-TF). For comparability the same RNN architecture (the dendPLRNN, Eq. 1) was used in all these comparisons.

Since in DS reconstruction, we are primarily interested in invariant and long-term properties of the underlying DS (Mikhaeil et al., 2022; Platt et al., 2023; Jiang et al., 2023; Hess et al., 2023; Platt et al., 2021), for evaluation we focused on measures that capture the geometrical structure in state space (D_{stsp}, a type of Kullback-Leibler divergence; Koppe et al. (2019b); Brenner et al. (2022); Hess et al.

(2023)) and the asymptotic temporal structure (evaluated either through the Hellinger distance, D_H, on power spectra (Mikhaeil et al., 2022; Hess et al., 2023), or auto-covariance functions for the ordinal and categorical data; Methods A.2). We also report mean-squared errors (MSE) to assess short-term ahead prediction (10-step-ahead L_2 prediction error (PE) for continuous and L_1 PE for ordinal data (Öğretir et al., 2022)). However, especially in chaotic systems, PEs are not suited for evaluating the system's longer-term behavior because of the exponential divergence of nearby trajectories (Wood, 2010; Mikhaeil et al., 2022). Details of all measures are provided in Methods A.2. As evidenced in Table 1, MTF outperforms all other possible model setups, and in particular the multimodal SVAE by large margins, reinforcing our point about the difficulty of learning an appropriate posterior in the absence of control-theoretic guidance of the training process (cf. Fig. A6).
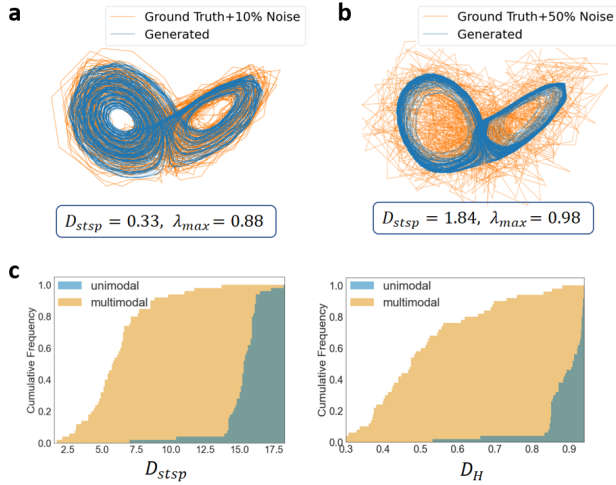
Figure 2. DS reconstruction from moderately (**a**) and heavily (**b-c**) distorted continuous observations (Gaussian observation noise of 10% and 50%, respectively, of the data variance) and other simultaneously provided observation modalities, sampled from a Lorenz-63 system. **a**: Freely generated example trajectories from a dendPLRNN ($M = 20, B = 10, K = 20, \tau = 10$) trained with MTF jointly on Gaussian (10% noise), ordinal, and count data. **b**: Same as **a** for a dendPLRNN trained by MTF on heavily distorted Gaussian (50% noise) and ordinal observations. Note that even in this case the butterfly wing structure of the Lorenz attractor was successfully captured. The maximum Lyapunov exponent ($\lambda_{\max}$) furthermore confirms the dendPLRNN-generated attractor is chaotic (for the GT Lorenz system, $\lambda_{\max} \approx 0.903$). **c**: Normalized cumulative histograms of geometrical attractor disagreement (D_{stsp}, left) and Hellinger distance (D_H, right) between reconstructed and ground-truth system for the same setting as in **b**.

To investigate multimodal data integration in a more challenging setting, we next tested a scenario where continuous observations from the Lorenz-63 system were heavily distorted by Gaussian noise with 50% of the data variance.

At the same time, ordinal observations with 8 variables divided into 7 ordered categories each, $o_{nt} \in \{1 \dots 7\}$, $n = 1 \dots 8$, were sampled using Eq. 14. We then trained the dendPLRNN via MTF once with, and once without, ordinal observations. Fig. 2**b** proves that with ordinal observations on board, DS reconstruction is, in principle, possible even under these challenging conditions. The cumulative histograms of the geometric measure D_{stsp} and the temporal measure D_H in Fig. 2**c** furthermore show that inclusion of ordinal observations significantly improves reconstructions.
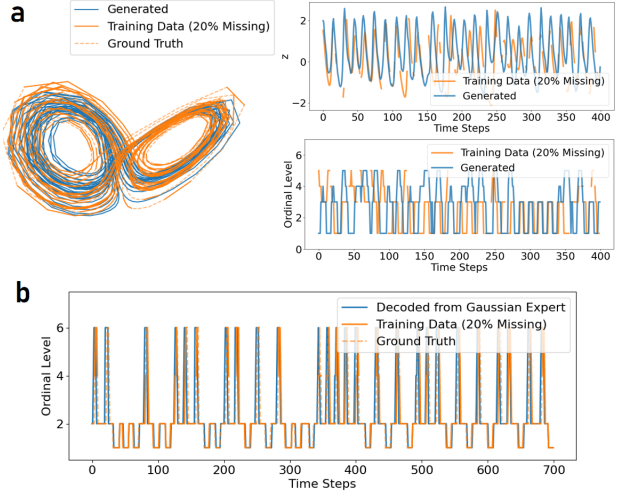
Figure 3. Cross-modal inference with missing observations, using a mixture-of-experts encoder. **a**: Reconstruction of the Lorenz-63 from Gaussian and ordinal observations, with 20% of data removed at random times chosen independently for each modality. **b**: Using only the Gaussian expert, the corresponding ordinal observations can be decoded almost perfectly, including at times missing in the ordinal training data. Dashed lines indicate sections with missing data in **a** and **b**.

Finally, MTF can easily handle missing data in one or more modalities, by simply dropping missing data points from the loss terms and using encoders like a mixture-of-experts (Appx. A.4), as illustrated in Fig. 3. As Fig. 3 highlights as well, MTF can also be employed for *cross-modal inference*, predicting outcomes for unobserved modalities through the joint encoder model.

3.2. DSR from Discrete Random Variables

Motivated by these results, we pushed the system even further and attempted DS reconstruction *solely from ordinal data* (created as above), i.e. completely omitting continuous observations. This is profoundly more challenging than the multimodal setting with Gaussian data, since the ordinal process considerably coarse-grains the underlying continuous dynamical process. Since in this case we do not have a direct linear mapping between ground truth state space and that of the trained dendPLRNN (which in the case of
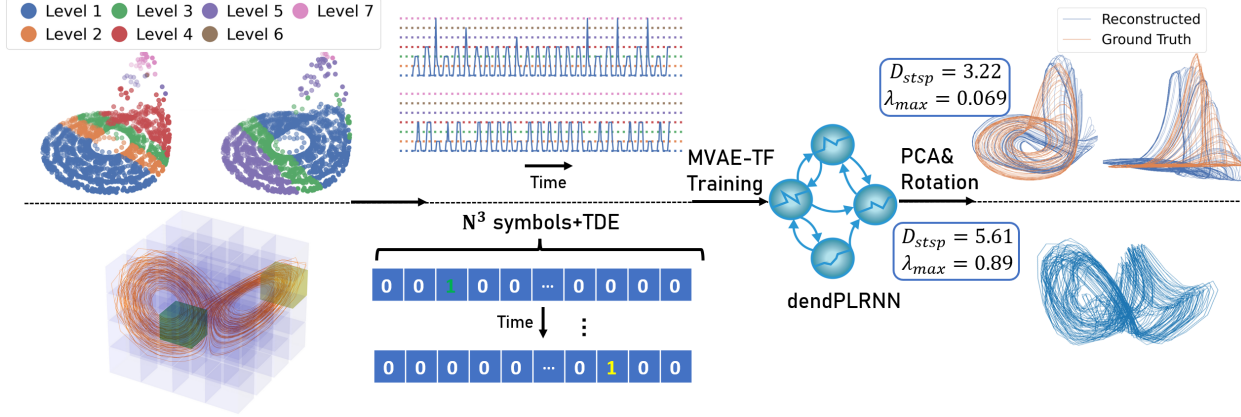
Figure 4. DS reconstruction from discrete observations by MTF ($M = 30, B = 15, K = 30, \tau = 10$). Top: Reconstruction of Rössler attractor from only ordinal time series. Bottom: DS reconstruction from symbolic coding of Lorenz attractor (see Fig. A9 for true and predicted class label probabilities). Ground truth systems and their ordinal/symbolic encoding are on the left, corresponding reconstructions on the right. Note that in both cases the topology and general geometry are preserved, and maximal Lyapunov exponents closely match those of the true systems (Rössler: $\lambda_{max}^{true} \approx 0.072$, Lorenz: $\lambda_{max}^{true} \approx 0.903$). TDE = temporal delay embedding.

Gaussian observations would simply be given by the linear decoder, Eq. 12), we construct one post-hoc by optimizing a linear operator given by a linear dimensionality reduction (PCA) concatenated with a geometry-preserving rotation operation (see Methods A.2). Fig. 4 (top) confirms that, using MTF, successful DS reconstruction is still possible under these conditions (see also Fig. A14 for a higher-dimensional example of reconstruction from only ordinal data).

Finally, we asked whether DS reconstruction might be feasible based on a purely *symbolic representation* of the dynamics. For this, 4^3 symbols were used corresponding to subregions defined by a $4 \times 4 \times 4$ grid superimposed on the attractor (this symbolic code was reduced further and then delay-embedded, see Appx. A.5 for full details). Fig. 4 (bottom) illustrates the general procedure, and also shows that, using MTF, successful DS reconstruction is – in principle – possible from just a symbolic (categorical) coding of the underlying chaotic attractor. This is to our knowledge the first time such a result has been shown. Comparable results could not be achieved by the multimodal SVAE or MS (see Table B4). Fig. A9**b** shows that even the maximum Lyapunov exponent λ_{max} of the system reconstructed with MTF from just the symbolic coding of the Lorenz attractor often comes close to the true value (see also Fig. A8).

3.3. DSR from Neuroimaging and Behavioral Data

We evaluated MTF's performance on real multimodal time series using a dataset of functional magnetic resonance imaging (fMRI) recordings from human subjects undertaking various cognitive tasks in an fMRI scanner (Koppe et al., 2014). The data consisted of continuous blood-oxygenation-level-dependent (BOLD) time series acquired from 20 subjects

Table 2. Comparison among multi-modal reconstruction methods for experimental fMRI+behavioral data. For each subject and training method, medians across 15 trained models were first obtained for each measure, which were then averaged across 20 subjects ($\pm$ SEM). SEM = standard error of the mean. X = value not accessible for this method. For abbreviations see Table 1.

Dataset	Method	$D_{stsp} \downarrow$	$D_H \downarrow$	PE $\downarrow$
	MTF	**0.55 ± 0.04**	**0.301 ± 0.007**	**1.21 ± 0.08**
	SVAE	1.9 ± 0.22	0.441 ± 0.019	2.34 ± 0.12
fMRI	BPTT	3.31 ± 0.8	0.52 ± 0.05	2.8 ± 0.15
	MS	1.06 ± 0.14	0.373 ± 0.012	X
	GVAE-TF	0.67 ± 0.06	0.335 ± 0.011	1.64 ± 0.07
	BPTT-TF	0.63 ± 0.03	0.312 ± 0.006	1.39 ± 0.05

at a sampling rate of $1/3$ Hz. The first principal component of BOLD activity from each of 20 different brain regions was selected for analysis. The individual time series were rather short ($T = 360$ per subject), posing a significant challenge for DS reconstruction. In addition to continuous BOLD activity, categorical time series of cognitive stage labels were given for each subject, corresponding to the five cognitive tasks that each subject went through repeatedly during the experiment ('Rest', 'Instruction', 'Choice Reaction Task [CRT]', 'Continuous Delayed Response Task [CDRT]' and 'Continuous Matching Task [CMT]')[4], which we accommodated via a multi-categorical decoder model, Eq. 16 in Methods A.1.

This is the same dataset and setup as considered in Kramer et al. (2022), which we use here for comparability, repeating the same type of analyses as performed in that previous

---

[4]Note that CRT, CDRT, and CMT did not differ in terms of presented stimuli or response types, but only in cognitive load, thus reflecting processes internal to the subject.
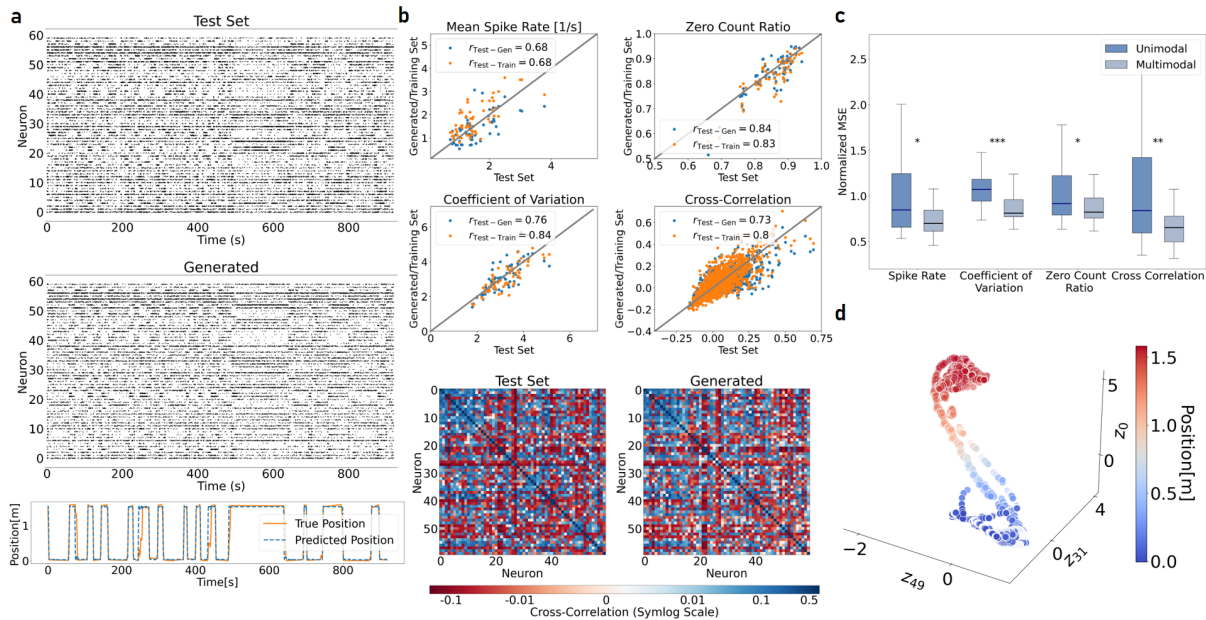
Figure 5. **a**: Example reconstructions of spike trains and spatial location of animal. Top: Spike train data from a test set not used for model training (topmost), and model-generated spike trains (below) simulated from a data-inferred initial condition. Bottom: True and model-predicted spatial position on the test set. **b**, top: Correlation of various spike train statistics between test set and model-generated data (blue), and – for comparison – between experimental training and test set data (orange): mean spike rate, zero count ratio, coefficient of variation, and correlation between cross-correlation coefficients. Diagonal gray lines are bisectrices, not regression lines. Bottom: Cross-correlation matrices among all neurons for the experimental test set (left) and model-generated spike data (right). **c**: Joint DSR from both spatial and neural data significantly improves reconstructions compared to just neural data alone ($*p < 0.05, ***p < 0.001$). **d**: DSR model latent space (shown is a subspace), illustrating how the latent dynamics is organized according to the animal's spatial position (color-coded).

study for our MTF. We first compared the performance of a dendPLRNN trained with MTF only on the BOLD signals to the performance when the same algorithm was additionally provided with the categorical information, optimizing a joint Gaussian and categorical likelihood (see Appx. Eq. 16). As shown in Fig. A2**a** and statistically confirmed by a paired t-test, according to D_{stsp} ($t_{19} = 2.45, p < .013$) and D_H ($t_{19} = 2.72, p < .007$), training the dendPLRNN with multimodal data significantly improved DS reconstruction compared to training on just the BOLD signals. These results confirm that even in this empirical situation categorical data can significantly help to improve DS reconstruction, as Kramer et al. (2022) observed previously for their algorithm.

Fig. A2**b** further shows an example where on a left-out test set simulated BOLD activity and model-generated task labels closely resembled those of the real subject. Fig. A2**c** investigates to what extent cross-modal links have been built by the trained system in its latent space by illustrating that after multimodal training, freely sampled latent activity trajectories from the dendPLRNN reflect the structure of the different tasks stages. Fig. A3 in Appx. A.5 provides generated time series for several example subjects. Table 2, finally, compares the DSR performance on the fMRI

and behavioral data for all the methods introduced in Sect. 3, with MTF substantially outperforming the multimodal SVAE from (Kramer et al., 2022), again highlighting the crucial differences between these approaches.

3.4. DSR from Neurophysiological and Location Data

As a second empirical test case, we used electrophysiological recordings of multiple single neurons from the CA1 region of the rodent hippocampus (Grosmark et al., 2016), a brain region involved in spatial navigation and memory (Buzsaki, 2006; Moser et al., 2017). During the experiments the rats traversed a 1.6 m long track along which the longitudinal position of the animal was recorded, treated as a continuously valued Gaussian 1d time series. Water rewards were provided at each of the two end points of the track, indicated to the model through brief external inputs. At the same time, for the data set on which we focused our analysis, the spiking activity (action potential time points) of 120 neurons was recorded via intracortical electrodes (of which 60 with reasonably high spiking rate were retained). Following common procedures in neuroscience (Zhou & Wei, 2020; Schneider et al., 2023), spike trains were segmented

into 200 ms bins, resulting in a multivariate count time series (for details, see Methods A.5). These were modeled with a negative-binomial decoder (Eq. 20), capturing the high dispersion better than Poisson or zero-inflated Poisson models.

Multimodal (position and spike count) data were split into a training and a test set of 4600 time steps each. Example true and model-generated spike trains for the test set are in Fig. 5a (see Fig. A11 for training set). Fig. 5b compares various standard spike train statistics like the mean firing rate, mean zero rate, coefficient of variation, and the neural cross-correlation matrix between experimental and model-generated data, iterated forward from one inferred initial state across the whole test set. The agreement in spiking statistics between model-simulated long-term behavior and test data was in fact as good as between different segments (training and test set) of the experimental data itself. Moreover, the predicted position on the track closely matched that of the real behavior for the test set (Fig. 5a). Using both data sources (spike trains and position) led to far better reconstructions across all spike train performance metrics, and overall more robust reconstructions, than when only spike count data were provided for model training (Mann-Whitney U-test across 50 trained models, $p < 0.025$ for all performance metrics; Fig. 5c). In the CA1 region, place cells are known to encode spatial information (O'Keefe & Nadel, 1978). Our results indicate that the MTF algorithm can leverage this information to build a joint latent model of the neural recordings and movement of the rat (Fig. 5d).

4. Conclusions

Here we introduced a novel training framework (MTF) for DSR from non-Gaussian and multimodal time series data. While DS reconstruction is meanwhile a large field in scientific machine learning (Brunton et al., 2016; Pathak et al., 2018; Lejarza & Baldea, 2022; Brenner et al., 2022; Hess et al., 2023; Chen & Wu, 2023), reconstruction based on discrete or multimodal data has hardly been addressed so far, although such scenarios are commonplace in many areas like medicine, neuroscience, or climate research. Here we utilize recent insights on guiding the training process by control signals, as in STF (Mikhaeil et al., 2022) and GTF (Hess et al., 2023), within a novel multimodal DSR framework. In our approach, a sparse TF signal is generated by an MVAE that translates different data modalities into a common latent code in order to guide a DSR model. Our general framework is very flexible and allows to easily swap DSR, encoder, or decoder models for the problem at hand. For diverse benchmarks and sampling conditions, MTF clearly outperforms various other training setups and models. Moreover, we found MTF to be quite robust performance-wise, and not very sensitive to precise hyper-parameter tuning.

We further highlighted on examples from neuroscience how the DSR model's latent space exposes meaningful relations between different data modalities (neural activity and behavior in our case). In contrast to other recently proposed methods for construing a common latent space from multiple data sources (Schneider et al., 2023), our method actually delivers a *generative model of the system dynamics*. A crucial difference to 'classical' multimodal SVAEs (Kramer et al., 2022) is that our algorithm allows to optimally control trajectory and gradient flows whilst training, a key ingredient for successful DS reconstruction (Mikhaeil et al., 2022; Hess et al., 2023; Eisenmann et al., 2023). One potentially interesting future direction may be engineering ways of incorporating control signals directly at the encoder level.

This way, MTF was even able, under certain conditions, to recover chaotic attractors from ordinal or categorical observations alone, including sensible estimates of their maximum Lyapunov exponent. This, to our knowledge, has never been shown before, and at first glance appears puzzling, as an ordinal or symbolic coding of the dynamics seems to remove much of the geometrical information. However, results in the field of symbolic dynamics ensure that under certain conditions a symbolic encoding of a DS preserves topological invariants and Lyapunov spectra (Osipenko, 2000; Hubler, 2012; McCullough et al., 2017; Zhang et al., 2017; Lind et al., 2021), and delay embedding theorems have been formulated for non-continuous signals like point processes (Sauer, 1994; 1995). While we foresee the major applications of MTF in truly multimodal settings, our observations here may also pave the way for interesting novel DSR applications to purely symbolic data. For instance, reconstructing dynamics from categorical consumer data or sequences of behavioral choices (on which classically reinforcement learning models would be trained) may become feasible, or even from language data (Elman, 1995), e.g. recovering the dynamics of underlying beliefs or values (Tsakalidis et al., 2022).

While here we presented first results on DS reconstruction for discrete data, whether and how much of the original state space topology of a data-generating DS can be recovered from non-continuous, non-Gaussian random variables remains an important topic for future theoretical and empirical research. Another open question remains under which conditions different modalities support (or interfere with) each other in the DSR process.

Software and Data

All code created is available at `https://github.com/DurstewitzLab/MTF`.

Acknowledgements

Impact Statement

This paper presents a novel approach in the context of scientific machine learning, with applications in fields like neuroscience. We believe the primary ethical implications of this approach are positive. While we can not specifically highlight any ethical concerns, given the broad scope of possible applications of our approach, potential misuses can not be a priori excluded.

References

Abarbanel, H. *Predicting the future: completing models of observed complex systems*. Springer, 2013.

Abarbanel, H. D., Creveling, D. R., Farsian, R., and Kostuk, M. Dynamical state and parameter estimation. *SIAM Journal on Applied Dynamical Systems*, 8(4):1341–1381, 2009.

Abarbanel, H. D. I. *The Statistical Physics of Data Assimilation and Machine Learning*. Cambridge University Press, Cambridge, 2022. ISBN 978-1-316-51963-9. doi: 10.1017/9781009024846.

Ahuja, C., Morency, L. P., et al. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, pp. 1–20, 2017.

Alligood, K. T., Sauer, T. D., and Yorke, J. A. *Chaos: An Introduction to Dynamical Systems*. Textbooks in Mathematical Sciences. Springer, 1996. ISBN 978-0-387-94677-1 978-0-387-22492-3. doi: 10.1007/b97589. URL http://link.springer.com/10.1007/b97589.

Amrani, E., Ben-Ari, R., Rotman, D., and Bronstein, A. Noise Estimation Using Density Estimation for Self-Supervised Multimodal Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35:6644–6652, May 2021. doi: 10.1609/aaai.v35i8.16822.

Antelmi, L., Ayache, N., Robert, P., and Lorenzi, M. Multi-channel stochastic variational inference for the joint analysis of heterogeneous biomedical data in alzheimer's disease. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pp. 15–23, Cham, 2018. Springer International Publishing. ISBN 978-3-030-02628-8.

Arribas, D., Zhao, Y., and Park, I. M. Rescuing neural spike train models from bad MLE. In *Advances in Neural Information Processing Systems*, volume 33, pp. 2293–2303. Curran Associates, Inc., 2020.

Bai, J., Wang, W., and Gomes, C. P. Contrastively disentangled sequential variational autoencoder. *Advances in Neural Information Processing Systems*, 34:10105–10118, 2021.

Baltrusaitis, T., Ahuja, C., and Morency, L.-P. Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):423–443, February 2019. ISSN 0162-8828. doi: 10.1109/TPAMI.2018.2798607. URL https://doi.org/10.1109/TPAMI.2018.2798607.

Bayer, J., Soelch, M., Mirchev, A., Kayalibay, B., and Smagt, P. v. d. Mind the Gap when Conditioning Amortised Inference in Sequential Latent-Variable Models. In *Proceedings of the Ninth International Conference on Learning Representations*, October 2021. URL https://openreview.net/forum?id=a2gqxKDvYys.

Bengio, Y., Simard, P., and Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994. ISSN 1045-9227. doi: 10.1109/72.279181.

Bhagwat, N., Viviano, J. D., Voineskos, A. N., Chakravarty, M. M., and Initiative, A. D. N. Modeling and prediction of clinical symptom trajectories in alzheimer's disease using longitudinal data. *PLOS Computational Biology*, 14(9):e1006376, 2018. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1006376. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1006376.

Bock, H. G. and Plitt, K. J. A multiple shooting algorithm for direct solution of optimal control problems*. *IFAC Proceedings Volumes*, 17(2):1603–1608, 1984. ISSN 1474-6670. doi: 10.1016/S1474-6670(17)61205-9. URL https://www.sciencedirect.com/science/article/pii/S1474667017612059.

Box, G. E. P. and Cox, D. R. An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 26(2):211–252, 1964. ISSN 0035-9246. URL https://www.jstor.org/stable/2984418. Publisher: Royal Statistical Society, Wiley.

Brenner, M., Hess, F., Mikhaeil, J. M., Bereska, L. F., Monfared, Z., Kuo, P.-C., and Durstewitz, D. Tractable dendritic RNNs for reconstructing nonlinear dynamical systems. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2292–2320. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/brenner22a.html.

Brunton, S. L., Proctor, J. L., and Kutz, J. N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences USA*, 113(15):3932–3937, 2016. ISSN 0027-8424. doi: 10.1073/pnas.1517384113. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4839439/.

Bury, T. M., Sujith, R. I., Pavithran, I., Scheffer, M., Lenton, T. M., Anand, M., and Bauch, C. T. Deep learning for early warning signals of tipping points. *Proceedings of the National Academy of Sciences*, 118(39):e2106140118, 2021. doi: 10.1073/pnas.2106140118. URL https://www.pnas.org/doi/10.1073/pnas.2106140118.

Buzsaki, G. *Rhythms of the Brain*. Oxford university press, 2006.

Chen, J. and Wu, K. Deep-osg: Deep learning of operators in semigroup. *Journal of Computational Physics*, 493:112498, 2023.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In Moschitti, A., Pang, B., and Daelemans, W. (eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179. URL https://aclanthology.org/D14-1179.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, December 2014. URL http://arxiv.org/abs/1412.3555. arXiv:1412.3555 [cs].

Datseris, G. Dynamicalsystems.jl: A julia software library for chaos and nonlinear dynamics. *Journal of Open Source Software*, 3(23):598, mar 2018. doi: 10.21105/joss.00598. URL https://doi.org/10.21105/joss.00598.

Dezfouli, A., Morris, R., Ramos, F. T., Dayan, P., and Balleine, B. Integrated accounts of behavioral and neuroimaging data using flexible recurrent neural network

models. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Durstewitz, D., Koppe, G., and Thurm, M. I. Reconstructing computational system dynamics from neural data with recurrent neural networks. *Nature Reviews. Neuroscience*, 24(11):693–710, November 2023. ISSN 1471-0048. doi: 10.1038/s41583-023-00740-7.

Eisenmann, L., Monfared, Z., Göring, N. A., and Durstewitz, D. Bifurcations and loss jumps in rnn training. *arXiv preprint arXiv:2310.17561*, 2023.

Elman, J. L. Language as a dynamical system. In *Mind as motion: Explorations in the dynamics of cognition*, pp. 195–225. The MIT Press, Cambridge, MA, US, 1995. ISBN 978-0-262-16150-3.

Fu, Y., Saab Jr, S., Ray, A., and Hauser, M. A Dynamically Controlled Recurrent Neural Network for Modeling Dynamical Systems. *arXiv:1911.00089 [cs, stat]*, October 2019. URL http://arxiv.org/abs/1911.00089. arXiv: 1911.00089.

Gauthier, D. J., Bollt, E., Griffith, A., and Barbosa, W. A. S. Next generation reservoir computing. *Nature Communications*, 12(1):5564, September 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-25801-2. URL https://www.nature.com/articles/s41467-021-25801-2. Number: 1 Publisher: Nature Publishing Group.

Gilpin, W. Chaos as an interpretable benchmark for forecasting and data-driven modelling. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2022. URL https://openreview.net/forum?id=enYjtbjYJrf.

Girin, L., Leglaive, S., Bie, X., Diard, J., Hueber, T., and Alameda-Pineda, X. Dynamical variational autoencoders: A comprehensive review. *Foundations and Trends® in Machine Learning*, 15(1):1–175, 2021. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000089. URL http://arxiv.org/abs/2008.12595.

Gower, J. C. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975. ISSN 1860-0980. doi: 10.1007/BF02291478. URL https://doi.org/10.1007/BF02291478.

Grosmark, A. D., Long, J., and Buzsáki, G. Recordings from hippocampal area ca1, pre, during and post novel spatial learning., March 2016.

Hess, F., Monfared, Z., Brenner, M., and Durstewitz, D. Generalized teacher forcing for learning chaotic dynamics. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Pro-*

ceedings of the 40th International Conference on Machine Learning, volume 202 of *Proceedings of Machine Learning Research*, pp. 13017–13049. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/hess23a.html.

Hinton, G. E. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, nov 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Hochreiter, S., Bengio, Y., Frasconi, P., Schmidhuber, J., et al. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies, 2001.

Hubler, A. Is symbolic dynamics the most efficient data compression tool for chaotic time series? *Complexity*, 17 (3):5–7, 2012.

Iakovlev, V., Yildiz, C., Heinonen, M., and Lähdesmäki, H. Latent Neural ODEs with Sparse Bayesian Multiple Shooting. September 2022. URL https://openreview.net/forum?id=moIlFZfj_1b.

Jiang, R., Lu, P. Y., Orlova, E., and Willett, R. Training neural operators to preserve invariant measures of chaotic attractors. *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023.

Jordana, A., Carpentier, J., and Righetti, L. Learning Dynamical Systems from Noisy Sensor Measurements using Multiple Shooting. *arXiv:2106.11712 [cs, stat]*, June 2021. URL http://arxiv.org/abs/2106.11712. arXiv: 2106.11712.

Kantz, H. and Schreiber, T. *Nonlinear time series analysis*, volume 7. Cambridge university press, 2004.

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations*, 2014. URL http://arxiv.org/abs/1312.6114.

Koppe, G., Gruppe, H., Sammer, G., Gallhofer, B., Kirsch, P., and Lis, S. Temporal unpredictability of a stimulus sequence affects brain activation differently depending on cognitive task demands. *NeuroImage*, 101:236–244, 2014. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2014.07.008.

Koppe, G., Guloksuz, S., Reininghaus, U., and Durstewitz, D. Recurrent neural networks in mobile sampling and intervention. *Schizophrenia Bulletin*, 45(2):272–276, 2019a. ISSN 1745-1701. doi: 10.1093/schbul/sby171.

Koppe, G., Toutounji, H., Kirsch, P., Lis, S., and Durstewitz, D. Identifying nonlinear dynamical systems via generative recurrent neural networks with applications to fMRI. *PLOS Computational Biology*, 15(8):e1007263, 2019b. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1007263. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1007263.

Kramer, D., Bommer, P. L., Tombolini, C., Koppe, G., and Durstewitz, D. Reconstructing nonlinear dynamical systems from multi-modal time series. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 11613–11633. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/kramer22a.html.

Lambert, D. Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, 34(1):1–14, 1992. ISSN 0040-1706. doi: 10.2307/1269547. URL https://www.jstor.org/stable/1269547. Publisher: [Taylor & Francis, Ltd., American Statistical Association, American Society for Quality].

Lejarza, F. and Baldea, M. Data-driven discovery of the governing equations of dynamical systems via moving horizon optimization. *Scientific Reports*, 12(1): 11836, July 2022. ISSN 2045-2322. doi: 10.1038/s41598-022-13644-w. URL https://www.nature.com/articles/s41598-022-13644-w. Number: 1 Publisher: Nature Publishing Group.

Lewis, J. E. and Glass, L. Nonlinear dynamics and symbolic dynamics of neural networks. *Neural Computation*, 4(5): 621–642, 1992. ISSN 0899-7667. doi: 10.1162/neco.1992.4.5.621. URL https://doi.org/10.1162/neco.1992.4.5.621.

Liang, M., Li, Z., Chen, T., and Zeng, J. Integrative data analysis of multi-platform cancer data with a multimodal deep learning approach. *IEEE/ACM transactions on computational biology and bioinformatics*, 12(4):928–937, 2015. ISSN 1557-9964. doi: 10.1109/TCBB.2014.2377729.

Liang, P. P., Zadeh, A., and Morency, L.-P. Foundations and recent trends in multimodal machine learning: Principles, challenges, and open questions. *arXiv preprint arXiv:2209.03430*, 2022.

Liddell, T. M. and Kruschke, J. K. Analyzing ordinal data with metric models: What could possibly go wrong? *Journal of Experimental Social Psychology*, 79:328–348, 2018. ISSN 0022-1031. doi: 10.1016/j.jesp.2018.08.

009. URL https://www.sciencedirect.com/science/article/pii/S0022103117307746.

Likert, R. A technique for the measurement of attitudes. *Archives of Psychology*, 22 140:55–55, 1932.

Lind, D. A., Lind, D., and Marcus, B. *An introduction to symbolic dynamics and coding*. Cambridge university press, 2021.

Lipkova, J., Chen, R. J., Chen, B., Lu, M. Y., Barbieri, M., Shao, D., Vaidya, A. J., Chen, C., Zhuang, L., Williamson, D. F. K., Shaban, M., Chen, T. Y., and Mahmood, F. Artificial intelligence for multimodal data integration in oncology. *Cancer Cell*, 40(10):1095–1110, October 2022. ISSN 1878-3686. doi: 10.1016/j.ccell.2022.09.012.

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Han, J. On the variance of the adaptive learning rate and beyond. In *Proceedings of the Eighth International Conference on Learning Representations*, 2020. URL https://iclr.cc/virtual_2020/poster_rkgz2aEKDr.html.

Lorenz, E. N. Deterministic nonperiodic flow. *Journal of atmospheric sciences*, 20(2):130–141, 1963.

Lusch, B., Kutz, J. N., and Brunton, S. L. Deep learning for universal linear embeddings of nonlinear dynamics. *Nat Commun*, 9(1):4950, December 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-07210-0. URL http://arxiv.org/abs/1712.09707. arXiv: 1712.09707.

Matilla-García, M., Morales, I., Rodríguez, J. M., and Ruiz Marín, M. Selection of embedding dimension and delay time in phase space reconstruction via symbolic dynamics. *Entropy*, 23(2):221, 2021. ISSN 1099-4300. doi: 10.3390/e23020221. URL https://www.mdpi.com/1099-4300/23/2/221. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

McCullagh, P. Regression models for ordinal data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 42(2):109–142, 1980. ISSN 0035-9246. URL https://www.jstor.org/stable/2984952.

McCullough, M., Sakellariou, K., Stemler, T., and Small, M. Regenerating time series from ordinal networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 27(3), 2017.

Meng, F., Richer, M., Tehrani, A., La, J., Kim, T. D., Ayers, P. W., and Heidar-Zadeh, F. Procrustes: A python library to find transformations that maximize the similarity between matrices. *Computer Physics Communications*, 276(108334):1–37, 2022. ISSN 0010-4655. doi: https://doi.org/10.1016/j.cpc.2022.108334.

URL https://www.sciencedirect.com/science/article/pii/S0010465522000522.

Mikhaeil, J. M., Monfared, Z., and Durstewitz, D. On the difficulty of learning chaotic dynamics with RNNs. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, October 2022. URL https://openreview.net/forum?id=-_AMpmyV0Ll.

Monfared, Z. and Durstewitz, D. Transformation of ReLU-based recurrent neural networks from discrete-time to continuous-time. In *Proceedings of the 37th International Conference on Machine Learning*, 2020. URL http://proceedings.mlr.press/v119/monfared20a.html.

Moser, E. I., Moser, M.-B., and McNaughton, B. L. Spatial representation in the hippocampal formation: a history. *Nature Neuroscience*, 20(11):1448–1464, November 2017. ISSN 1546-1726. doi: 10.1038/nn.4653. URL https://www.nature.com/articles/nn.4653. Number: 11 Publisher: Nature Publishing Group.

O'Keefe, J. and Nadel, L. *The Hippocampus as a Cognitive Map*. Oxford University Press, Oxford, UK, 1978. URL http://www.cognitivemap.net/. Publication Title: Oxford University Press: Oxford, UK. (1978).

OpenAI, R. Gpt-4 technical report. *arXiv*, pp. 2303–08774, 2023.

Osipenko, G. Spectrum of a dynamical system and applied symbolic dynamics. *Journal of Mathematical Analysis and Applications*, 252(2):587–616, 2000.

Patel, D. and Ott, E. Using machine learning to anticipate tipping points and extrapolate to post-tipping dynamics of non-stationary dynamical systems. *Chaos (Woodbury, N.Y.)*, 33(2):023143, February 2023. ISSN 1089-7682. doi: 10.1063/5.0131787.

Pathak, J., Hunt, B., Girvan, M., Lu, Z., and Ott, E. Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach. *Phys. Rev. Lett.*, 120(2):024102, 2018. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.120.024102. URL https://link.aps.org/doi/10.1103/PhysRevLett.120.024102.

Platt, J. A., Wong, A., Clark, R., Penny, S. G., and Abarbanel, H. D. Robust forecasting using predictive generalized synchronization in reservoir computing. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(12), 2021.

Platt, J. A., Penny, S. G., Smith, T. A., Chen, T.-C., and Abarbanel, H. D. I. A Systematic Exploration of Reservoir Computing for Forecasting Complex Spatiotemporal Dynamics, January 2022. URL http://arxiv.org/abs/2201.08910. arXiv:2201.08910 [cs].

Platt, J. A., Penny, S. G., Smith, T. A., Chen, T.-C., and Abarbanel, H. D. Constraining chaos: Enforcing dynamical invariants in the training of reservoir computers. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 33 (10), 2023.

Qian, S., Chou, C.-A., and Li, J.-S. Deep multi-modal learning for joint linear representation of nonlinear dynamical systems. *Scientific reports*, 12(1):12807, 2022.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763. PMLR, July 2021. URL https://proceedings.mlr.press/v139/radford21a.html. ISSN: 2640-3498.

Raissi, M., Perdikaris, P., and Karniadakis, G. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, February 2019. ISSN 00219991. doi: 10.1016/j.jcp.2018.10.045. URL https://linkinghub.elsevier.com/retrieve/pii/S0021999118307125.

Ramchandran, S., Tikhonov, G., Kujanpaa, K., Koskinen, M., and Lahdesmaki, H. Longitudinal variational autoencoder. *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pp. 11, 2021.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In *Proceedings of the 31st International Conference on Machine Learning*, 2014. URL http://arxiv.org/abs/1401.4082.

Rössler, O. E. An equation for continuous chaos. *Physics Letters A*, 57(5):397–398, 1976. ISSN 0375-9601. doi: 10.1016/0375-9601(76)90101-8. URL https://www.sciencedirect.com/science/article/pii/0375960176901018.

Sauer, T. Reconstruction of dynamical systems from inter-spike intervals. *Physical Review Letters*, 72(24):3811, 1994.

Sauer, T. Interspike interval embedding of chaotic signals. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 5(1):127–132, 1995.

Schmidt, D., Koppe, G., Monfared, Z., Beutelspacher, M., and Durstewitz, D. Identifying nonlinear dynamical systems with multiple time scales and long-range dependencies. In *Proceedings of the 9th International Conference on Learning Representations*, 2021. URL http://arxiv.org/abs/1910.03471.

Schneider, S., Lee, J. H., and Mathis, M. W. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, May 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06031-6. URL https://www.nature.com/articles/s41586-023-06031-6. Number: 7960 Publisher: Nature Publishing Group.

Shi, Y., Siddharth, N., Paige, B., and Torr, P. H. S. Variational mixture-of-experts autoencoders for multi-modal deep generative models. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 15718–15729. Curran Associates Inc., 2019.

Shi, Y., Paige, B., Torr, P., and N, S. Relating by Contrasting: A Data-efficient Framework for Multimodal Generative Models. In *Proceedings of the Ninth International Conference on Learning Representations*, October 2020. URL https://openreview.net/forum?id=vhKe9UFbrJo.

Steyaert, S., Pizurica, M., Nagaraj, D., Khandelwal, P., Hernandez-Boussard, T., Gentles, A. J., and Gevaert, O. Multimodal data fusion for cancer biomarker discovery with deep learning. *Nature Machine Intelligence*, 5(4): 351–362, 2023.

Sutter, T. M., Daunhawer, I., and Vogt, J. E. Generalized Multimodal ELBO. In *Proceedings of the Ninth International Conference on Learning Representations*, October 2020. URL https://openreview.net/forum?id=5Y21V0RDBV.

Talathi, S. S. and Vartak, A. Improving performance of recurrent neural network with relu nonlinearity. In *Proceedings of the 4th International Conference on Learning Representations*, 2016. URL http://arxiv.org/abs/1511.03771.

Tsakalidis, A., Chim, J., Bilal, I. M., Zirikly, A., Atzil-Slonim, D., Nanni, F., Resnik, P., Gaur, M., Roy, K., Inkster, B., Leintz, J., and Liakata, M. Overview of the CLPsych 2022 Shared Task: Capturing Moments of Change in Longitudinal User Posts. In Zirikly, A., Atzil-Slonim, D., Liakata, M., Bedrick, S., Desmet, B., Ireland, M., Lee, A., MacAvaney, S., Purver, M., Resnik, R., and Yates, A. (eds.), *Proceedings of the Eighth Workshop on Computational Linguistics and Clinical Psychology*, pp. 184–198, Seattle, USA, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/

2022.clpsych-1.16. URL https://aclanthology.org/2022.clpsych-1.16.

Tziperman, E., Scher, H., Zebiak, S. E., and Cane, M. A. Controlling spatiotemporal chaos in a realistic el niño prediction model. *Phys. Rev. Lett.*, 79: 1034–1037, Aug 1997. doi: 10.1103/PhysRevLett. 79.1034. URL https://link.aps.org/doi/10.1103/PhysRevLett.79.1034.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Verzelli, P., Alippi, C., and Livi, L. Learn to synchronize, synchronize to learn. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(8), 2021.

Vlachas, P. R. and Koumoutsakos, P. Learning from predictions: Fusing training and autoregressive inference for long-term spatiotemporal forecasts. *arXiv preprint arXiv:2302.11101*, 2023.

Vlachas, P. R., Byeon, W., Wan, Z. Y., Sapsis, T. P., and Koumoutsakos, P. Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proc. R. Soc. A.*, 474(2213):20170844, 2018. ISSN 1364-5021, 1471-2946. doi: 10.1098/rspa.2017.0844. URL https://royalsocietypublishing.org/doi/10.1098/rspa.2017.0844.

Vlachas, P. R., Pathak, J., Hunt, B. R., Sapsis, T. P., Girvan, M., Ott, E., and Koumoutsakos, P. Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics. *Neural Networks*, 126:191–217, 2020.

Vlachas, P. R., Arampatzis, G., Uhler, C., and Koumoutsakos, P. Multiscale simulations of complex systems by learning their effective dynamics. *Nature Machine Intelligence*, 4(4):359–366, 2022.

Vogt, R., Puelma Touzel, M., Shlizerman, E., and Lajoie, G. On lyapunov exponents for RNNs: Understanding information propagation using dynamical systems tools. *Frontiers in Applied Mathematics and Statistics*, 8, 2022. ISSN 2297-4687. URL https://www.frontiersin.org/articles/10.3389/fams.2022.818799.

Voss, H. U., Timmer, J., and Kurths, J. Nonlinear dynamical system identification from uncertain and indirect measurements. *International Journal of Bifurcation and Chaos*, 14(6):1905–1933, 2004. ISSN 0218-1274. doi: 10.1142/S0218127404010345. URL

https://www.worldscientific.com/doi/abs/10.1142/S0218127404010345.

Warner, E., Lee, J., Hsu, W., Syeda-Mahmood, T., Kahn, C., and Rao, A. Multimodal machine learning for clinically-assistive imaging-based biomedical applications. *arXiv preprint arXiv:2311.02332*, 2023.

Winship, C. and Mare, R. D. Regression models with ordinal variables. *American Sociological Review*, 49 (4):512–525, 1984. ISSN 0003-1224. doi: 10.2307/2095465. URL https://www.jstor.org/stable/2095465. Publisher: [American Sociological Association, Sage Publications, Inc.].

Wolf, A., Swift, J. B., Swinney, H. L., and Vastano, J. A. Determining lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*, 16(3):285–317, 1985. ISSN 0167-2789. doi: 10.1016/0167-2789(85)90011-9. URL https://www.sciencedirect.com/science/article/pii/0167278985900119.

Wood, S. N. Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466(7310):1102–1104, August 2010. ISSN 1476-4687. doi: 10.1038/nature09319. URL https://www.nature.com/articles/nature09319. Number: 7310 Publisher: Nature Publishing Group.

Wu, M. and Goodman, N. Multimodal generative models for scalable weakly-supervised learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 5580–5590, Red Hook, NY, USA, December 2018. Curran Associates Inc.

Xu, P., Zhu, X., and Clifton, D. A. Multimodal Learning With Transformers: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (10):12113–12132, October 2023. ISSN 0162-8828. doi: 10.1109/TPAMI.2023.3275156. URL https://www.computer.org/csdl/journal/tp/2023/10/10123038/1N3MioQlClW. Publisher: IEEE Computer Society.

Yang, L., Sun, X., Hamzi, B., Owhadi, H., and Xie, N. Learning dynamical systems from data: A simple cross-validation perspective, part v: Sparse kernel flows for 132 chaotic dynamical systems. *arXiv preprint arXiv:2301.10321*, 2023.

Zhang, J., Zhou, J., Tang, M., Guo, H., Small, M., and Zou, Y. Constructing ordinal partition transition networks from multivariate time series. *Scientific reports*, 7(1):7795, 2017.

Zhou, D. and Wei, X.-X. Learning identifiable and interpretable latent models of high-dimensional neural activity

using pi-vae. *Advances in Neural Information Processing Systems*, 33:7234–7247, 2020.

Öğretir, M., Ramchandran, S., Papatheodorou, D., and Lähdesmäki, H. A Variational Autoencoder for Heterogeneous Temporal and Longitudinal Data. In *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1522–1529, December 2022. doi: 10.1109/ICMLA55696.2022. 00239. URL https://ieeexplore.ieee.org/document/10069449.

A. Methods

A.1. MTF framework

The MTF framework consists of four major components, 1) a DSR model for approximating the flow of the underlying, observed DS, 2) a set of modality-specific decoder models that capture the statistical particularities of the observed time series, 3) an MVAE for integrating all different data modalities in order to produce a joint latent code which can be used for TF, and 4) the actual MTF training algorithm. Here we describe each of these components in turn.

DSR model   While our framework is generic and works with any type of DSR model, here we used the dendPLRNN, Eq. 1, recently introduced specifically for DSR (Brenner et al., 2022) (see Appx. Table B3 for comparison with other RNN models). A major advantage of the dendPLRNN and related formulations is that it provides some degree of analytical tractability, which is of major importance in scientific settings where we are usually interested in analyzing the data-inferred model further as a formal surrogate for the real system. For example, for the dendPLRNN, we have efficient algorithms for precisely locating all fixed points, cycles, and bifurcation manifolds (Eisenmann et al., 2023), and it allows for translation into an equivalent continuous-time representation (Monfared & Durstewitz, 2020), which further eases certain types of DS analysis.

Decoder models   To infer the latent process dendPLRNN jointly from multiple data modalities, it is connected to different decoder models that take the distinct distributional properties of each modality into account, Eqs. 2-4.

For *normally distributed* data, the decoder may take the simple linear Gaussian form

$$\boldsymbol{x}_t = \boldsymbol{B}\boldsymbol{z}_t + \boldsymbol{\eta}_t, \tag{12}$$

with factor loading matrix $\boldsymbol{B} \in \mathbb{R}^{N \times M}$, and Gaussian observation noise $\boldsymbol{\eta}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Gamma})$ with covariance $\boldsymbol{\Gamma} \in \mathbb{R}^{N \times N}$ (assumed to be diagonal here).

Ordinal data, in contrast, are not associated with a metric space, but there is a natural ordering between variables, as e.g. in survey data in economy or psychology commonly assessed through Likert scales (Likert, 1932). Treating ordinal data as metric can lead to a variety of problems, as pointed out in (Liddell & Kruschke, 2018). Ordinal observations are coupled to latent states via a generalized linear model (McCullagh, 1980). Here, specifically, we assume that the ordinal observations $\boldsymbol{o}_t$ are derived from an underlying unobserved continuous variable u_{it}, which is linked to the latent states $\boldsymbol{z}_t$ via a linear model

$$u_{it} = \boldsymbol{\beta}_i^T \boldsymbol{z}_t + \epsilon_{it}, \tag{13}$$

where $\boldsymbol{\beta}_i^T \in \mathbb{R}^M$ are the model parameters and ϵ_{it} is an independently distributed noise term. The distributional assumptions about the noise term ϵ_{it} determine which link function to use. A Gaussian assumption leads to an ordered probit model, while a logistic assumption leads to an ordered logit model (Winship & Mare, 1984). While both models lead to similar results, we found the ordered logit model to work slightly better in practice, and hence we focus on it here. Inverting the link function leads to an expression for the cumulative probabilities:

$$p\left(o_{it} \leq k | \boldsymbol{z}_t\right) = \frac{\exp\left(\beta_{ik}^0 - \boldsymbol{\beta}_i^T \boldsymbol{z}_t\right)}{1 + \exp\left(\beta_{ik}^0 - \boldsymbol{\beta}_i^T \boldsymbol{z}_t\right)}. \tag{14}$$

The probability masses $p\left(o_{it} = k | \boldsymbol{z}_t\right)$ follow from the cumulative distribution via $p\left(o_{it} = k | \boldsymbol{z}_t\right) = p\left(o_{it} \leq k | \boldsymbol{z}_t\right) - p\left(o_{it} \leq k - 1 | \boldsymbol{z}_t\right)$, from which we can compute the log-likelihood as

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{O}|\boldsymbol{Z}) = \sum_i^N \sum_t^T \sum_k^K [o_{it} = k] \log p\left(o_{it} = k | \boldsymbol{z}_t\right). \tag{15}$$

Categorical observations are, like ordinal observations, not associated with a metric space, and in contrast to ordinal data, there is also no natural ordering between variable values. To couple categorical observations to the latent states, we employed

the natural link function given by

$$\pi_i = \frac{\exp\left(\boldsymbol{\beta}_i^T \mathbf{z}_t\right)}{1 + \sum_{j=1}^{K-1} \exp\left(\boldsymbol{\beta}_j^T \mathbf{z}_t\right)} \quad \forall i \in \{1 \ldots K-1\} \tag{16}$$

$$\pi_K = \frac{1}{1 + \sum_{j=1}^{K-1} \exp\left(\boldsymbol{\beta}_j^T \mathbf{z}_t\right)}$$

Here, the parameters $\boldsymbol{\beta}_i \in \mathbb{R}^{M \times 1}$ constitute the respective regression weights for category $i = 1 \ldots K-1$, with the total probability over all categories $\sum_{i=1}^{K} \pi_i = 1$. This leads to the following log-likelihood for the categories:

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{O}|\boldsymbol{Z}) = \sum_i^N \sum_t^T \sum_k^K [o_{it} = k] \log\left(\frac{\exp\left(\boldsymbol{\beta}_k^T \boldsymbol{z}t\right)}{1 + \sum_{j=1}^{K-1} \exp\left(\boldsymbol{\beta}_j^T \boldsymbol{z}_t\right)}\right)$$

where $[o_{it} = k]$ is an indicator function that is 1 if the observation o_{it} belongs to category k and 0 otherwise.

For *count observations* $\{\boldsymbol{c}_t\}_{t=1}^T$, with $\boldsymbol{c}_t = (c_{1t}, \ldots, c_{Lt})^T$, we tried three different decoder models. First, a standard Poisson model,

$$p_{\boldsymbol{\theta}}\left(c_{lt}|\boldsymbol{z}_t\right) = \frac{\lambda_{lt}^{c_{lt}}}{c_{lt}!} e^{-\lambda_{lt}}. \tag{17}$$

These probabilities are related to the latent states via the log-link function, $\log \lambda_{lt} = \gamma_0^{(l)} + \sum_{m=1}^M \gamma_m^{(l)} z_{mt}$, where $\boldsymbol{\gamma}^{(l)}$ is a vector of coefficients. Thus, $\lambda_{lt} = e^{\gamma_0^{(l)} + \boldsymbol{\gamma}^{(l)} \boldsymbol{z_t}}$ is the expected count for the lth observation at time t. The total log-likelihood for observed counts $\boldsymbol{C}$ is then given by:

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{C}|\boldsymbol{Z}) = \sum_{t=1}^T \sum_{l=1}^L \left[c_{lt} \log \lambda_{lt} - \lambda_{lt} - \log(c_{lt}!)\right], \tag{18}$$

Alternatively, we tested a zero-inflated Poisson (ZIP) observation model (Lambert, 1992). The ZIP model is designed for count data that has an excess of zero counts compared to what would be expected from a traditional Poisson distribution, and thus can naturally handle overdispersion often observed in real data, such as the spike trains displayed in Fig. 5. The ZIP model addresses this by combining a binary process that determines whether a count is zero (with probability π_{lt}) with a Poisson distribution. The combined probability of an observed count given both processes is then given by

$$p_{\boldsymbol{\theta}}\left(c_{lt} \mid \boldsymbol{z}_t\right) = \begin{cases} \pi_{lt} + (1 - \pi_{lt}) e^{-\lambda_{lt}} & \text{if } c_{lt} = 0 \\ (1 - \pi_{lt}) \frac{\lambda_{lt}^{c_{lt}}}{c_{lt}!} e^{-\lambda_{lt}} & \text{if } c_{lt} > 0 \end{cases} \tag{19}$$

The probabilities are connected to the latent states via a logit and a log-link function $\log \frac{\pi_{lt}}{1-\pi_{lt}} = \beta_0^{(l)} + \sum_{m=1}^M \beta_m^{(l)} z_{mt}$ and $\log \lambda_{lt} = \gamma_0^{(l)} + \sum_{m=1}^M \gamma_m^{(l)} z_{mt}$, where $\gamma^{(l)}$ and $\beta^{(l)}$ are coefficient vectors. Thus, $\lambda_{lt} = e^{\gamma^{(l)} z_t}$ is the expected count for the lth observation and $\pi_{lt} = \frac{e^{\beta^{(l)} z_t}}{1 + e^{\beta^{(l)} z_t}}$ is the probability of observing a zero.

Finally, we tested a negative binomial model given by

$$p_{\boldsymbol{\theta}}\left(c_{lt}|\boldsymbol{z}_t\right) = \frac{\Gamma(c_{lt} + \phi_l)}{\Gamma(\phi_l) c_{lt}!} \left(\frac{\phi_l}{\mu_{lt} + \phi_l}\right)^{\phi_l} \left(\frac{\mu_{lt}}{\mu_{lt} + \phi_l}\right)^{c_{lt}}, \tag{20}$$

where μ_{lt} is the mean count and ϕ_l the dispersion parameter of the negative binomial distribution for the lth observation at time t. As for the Poisson model, we use a log-link function, $\log \mu_{lt} = \gamma_0^{(l)} + \sum_{m=1}^M \gamma_m^{(l)} z_{mt}$, with $\boldsymbol{\gamma}^{(l)}$ a vector of coefficients, and z_{mt} the mth latent variable at time t. Properly accounting for dispersion significantly improved the modeling of the spike counts. For the evaluation we used the the likelihood function implemented in the `torch.distributions.NegativeBinomial` class from the PyTorch library.

Multimodal variational autoencoder The MVAE consists of a set of decoder models, $p_{\boldsymbol{\theta}}(\boldsymbol{Y}|\tilde{\boldsymbol{Z}})$, shared with the DSR model, with examples given in Eqs. 6-8 and above, a prior model $p_{\boldsymbol{\theta}}(\tilde{\boldsymbol{Z}})$ instantiated by the DSR model, see Eq. 10, and the encoder (approximate posterior) $q_{\boldsymbol{\phi}}(\tilde{\boldsymbol{Z}}|\boldsymbol{Y})$, where a sample from this distribution provides the TF signal (see also Fig. 1). This setup is modular and all model components may be flexibly chosen. We assume conditional independence (given latent states) of the observations, such that the likelihood terms related to the observations simply sum up, i.e. $\log p_{\boldsymbol{\theta}}(\boldsymbol{Y}|\tilde{\boldsymbol{Z}}) = \sum_{t=1}^{T} (\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_t|\tilde{\boldsymbol{z}}_t) + \log p_{\boldsymbol{\theta}}(\boldsymbol{o}_t|\tilde{\boldsymbol{z}}_t) + \log p_{\boldsymbol{\theta}}(\boldsymbol{c}_t|\tilde{\boldsymbol{z}}_t))$ for the decoders specified in Eqs. 6-8 (with parameters shared with Eqs. 2-4).

For the encoder part, as the latent dendPLRNN process itself is conditionally Gaussian, we make a Gaussian assumption for the variational density $q_{\boldsymbol{\phi}}(\tilde{\boldsymbol{Z}}|\boldsymbol{Y}) = \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{Y}), \boldsymbol{\Sigma}_{\boldsymbol{\phi}}(\boldsymbol{Y}))$ to approximate the true posterior $p(\tilde{\boldsymbol{Z}}|\boldsymbol{Y})$, where mean and covariance are functions of the data. We further use the common mean field approximation to factorize $q_{\boldsymbol{\phi}}(\tilde{\boldsymbol{Z}}|\boldsymbol{Y})$ across time (Girin et al., 2021). The mean $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{Y})$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{Y})$ of the approximate posterior are parameterized by one-dimensional (across time) convolutional layers (Brenner et al., 2022), using observations within the window defined by the input kernel. For the mean, we use a 4-layer stack of convolutional layers with decreasing kernel sizes $11, 8, 5, 3$, while the diagonal covariance was parameterized by a single convolutional layer with a kernel size of 11. We also tested various other choices for the encoder, as discussed in Appx. A.4.

Multimodal teacher forcing Once we have a mechanism for generating a multimodal TF signal, our framework can accommodate any form of TF training scheme, like STF (Mikhaeil et al., 2022) or GTF (Hess et al., 2023). Consider an observed (multimodal) time series $\boldsymbol{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_T\}$ generated by a DS we would like to reconstruct. From this we create a control series $\hat{\boldsymbol{Z}} = \{\hat{\boldsymbol{z}}_1, \hat{\boldsymbol{z}}_2, \dots, \hat{\boldsymbol{z}}_T\}$, $\hat{\boldsymbol{z}}_t \in \mathbb{R}^M$. In STF or GTF, one obtains $\hat{\boldsymbol{Z}}$ from inverting the decoder model, projecting observations into the DSR model's latent space. Here, in contrast, we equate the control series with the latent code $\tilde{\boldsymbol{Z}} = \{\tilde{\boldsymbol{z}}_1, \dots, \tilde{\boldsymbol{z}}_T\}$, $\tilde{\boldsymbol{z}}_t \in \mathbb{R}^K$, produced by the MVAE. For the present evaluation we focused on a specific form of STF (Brenner et al., 2022), where the first K latent states are replaced by the control states $\hat{z}_{k,l\tau+1} = \tilde{z}_{k,l\tau+1}$, $k \leq K$, at times $l\tau + 1$, $l \in \mathbb{N}_0$, with forcing interval $\tau \geq 1$, while the remaining latent states, $\hat{z}_{k,l\tau+1} = z_{k,l\tau+1}$, $k > K$, remain unaffected. Defining $\mathcal{F} = \{l\tau + 1\}_{l \in \mathbb{N}_0}$, the dendPLRNN updates can thus be written as

$$\boldsymbol{z}_{t+1} = \begin{cases} \text{dendPLRNN}(\hat{\boldsymbol{z}}_t) & \text{if } t \in \mathcal{F} \\ \text{dendPLRNN}(\boldsymbol{z}_t) & \text{else} \end{cases} \tag{21}$$

For the initial condition $\boldsymbol{z}_1$ of the latent process, if $K < M$, the $M - K$ remaining states are randomly sampled from a standard normal distribution. As shown in Mikhaeil et al. (2022), the best tradeoff between exploding gradients and capturing relevant long-term dependencies is achieved when choosing the forcing interval τ according to the system's maximal Lyapunov exponent (predictability time). However, for non-continuous data (like counts), this cannot readily be determined, in which case the forcing interval τ may be regarded as a hyper-parameter determined via grid search (see also Appx. Fig. A7 for example reconstructions for different choices of τ).

To ensure that the dendPLRNN and MVAE can share all decoder model parameters, only the first $K \leq M$ states of the generated latent trajectory $\boldsymbol{Z}$ (using the generated states prior to forcing) are then used to compute the modality-specific negative log-likelihoods for the observed multimodal time series,

$$\mathcal{L}_{\text{DSR}} = -\sum_{t=1}^{T}(\log p_{\boldsymbol{\theta}}(\boldsymbol{x_t}|\boldsymbol{z}_{1:K,t}) + \log p_{\boldsymbol{\theta}}(\boldsymbol{o}_t|\boldsymbol{z}_{1:K,t}) + \log p_{\boldsymbol{\theta}}(\boldsymbol{c}_t|\boldsymbol{z}_{1:K,t})), \tag{22}$$

using the decoder models from Eqs. 2-4.

Likewise, to connect the latent codes of the MVAE and the RNN, the model prior of the MVAE is instantiated through the RNN by taking $\boldsymbol{\mu}_t = \boldsymbol{z}_{1:K,t}$, i.e., only the first K states of the generated latent sequence $\{\boldsymbol{z}_t\}$. As the initial state $\boldsymbol{z}_1$ is estimated directly from the encoded state $\tilde{\boldsymbol{z}}_1$, the term for $t = 1$ evaluates to zero. Setting $L = 1$, the *consistency loss*, Eq. 10, between encoded and generated latent state paths thus comes down to

$$\mathcal{L}_{\text{con}} = \frac{1}{2}\sum_{t=2}^{T}\big(\log |\boldsymbol{\Sigma}| + (\tilde{\boldsymbol{z}}_t - \boldsymbol{z}_{1:K,t})^\top \boldsymbol{\Sigma}^{-1}(\tilde{\boldsymbol{z}}_t - \boldsymbol{z}_{1:K,t})\big). \tag{23}$$

The total MTF loss is hence given by the loss in Eq. 23 that ensures consistency between the latent codes of the MVAE and DSR model, the DSR model loss from the likelihoods of the observed time series $\boldsymbol{Y}$ given the predicted latent path $\boldsymbol{Z}$

(Eq. 22), and the remaining terms from the ELBO, Eq. 9 (noting that $\mathcal{L}_{\text{con}} = -\mathbb{E}_{q_\phi}[\log p_\theta(\tilde{\boldsymbol{Z}})]$):

$$\mathcal{L}_{\text{MTF}} = -\mathbb{E}_{q_\phi}[\log p_\theta(\boldsymbol{Y}|\tilde{\boldsymbol{Z}})] - \mathbb{H}_{q_\phi}(\tilde{\boldsymbol{Z}}|\boldsymbol{Y}) + \mathcal{L}_{\text{con}} + \mathcal{L}_{\text{DSR}} \tag{24}$$

To train the dendPLRNN with MTF, RAdam (Liu et al., 2020) was used with a learning rate scheduler that iteratively reduced the learning rate from 10^{-3} to 10^{-5} during training. For each epoch, we randomly sampled sequences of length $T_{seq} = 300$ from the total training data with a batch size of 16, except for the fMRI data, where we chose $T_{seq} = 72$ due to the short overall length of the data ($T = 360$ per subject). The network weights $\boldsymbol{A}$, $\boldsymbol{W}$ and $\boldsymbol{h}$ from Eq. 1 were initialized as suggested in Talathi & Vartak (2016). Expansion term thresholds $\{\boldsymbol{h}_b\}$ and input matrix entries $[\boldsymbol{U}]_{ij}$ were drawn from a standard normal distribution, while dendritic slopes $\alpha_b \sim \mathcal{U}(-B^{-\frac{1}{2}}, B^{-\frac{1}{2}})$ were initialized uniformly.

A.2. Performance measures

Geometrical measure To assess the (dis-)agreement D_{stsp} between the data distribution $p_{\text{true}}(\boldsymbol{x})$ and the model-generated distribution $p_{\text{gen}}(\boldsymbol{x}|\boldsymbol{z})$ across *state space*, $\hat{p}_{\text{true}}(\boldsymbol{x})$ and $\hat{p}_{\text{gen}}(\boldsymbol{x}|\boldsymbol{z})$ are estimated by sampling 100 trajectories with randomly drawn initial conditions and 1000 time steps each. Transients are removed from each sampled trajectory to ensure that the (occupation) measure is evaluated on the limit set. The match between distributions is then approximated by binning the state space into discrete bins (Koppe et al., 2019b):

$$D_{\text{stsp}}\left(p_{\text{true}}(\boldsymbol{x}), p_{\text{gen}}(\boldsymbol{x}|\boldsymbol{z})\right) \approx \sum_{k=1}^{K} \hat{p}_{\text{true}}^{(k)}(\boldsymbol{x}) \log\left(\frac{\hat{p}_{\text{true}}^{(k)}(\boldsymbol{x})}{\hat{p}_{\text{gen}}^{(k)}(\boldsymbol{x}|\boldsymbol{z})}\right). \tag{25}$$

Here, K is the total number of bins. A range of $2\times$ the data standard deviation on each dimension was partitioned into m bins, leading to a total of $K = m^N$ bins, where N is the dimension of the ground truth system. Due to the exponential scaling of the number of bins with system dimensionality, for the 6-dimensional Lewis-Glass network model and the fMRI data we instead used an approximation of $p_{\text{true}}(\boldsymbol{x})$ and $p_{\text{gen}}(\boldsymbol{x}|\boldsymbol{z})$ based on Gaussian mixture models placed along trajectories, as described in Brenner et al. (2022).

Geometric reconstruction measure in the absence of continuous observations If the underlying DS was observed only through time series of discrete random variables, we lack a direct mapping between the true and reconstructed continuous state spaces. In this case, to still assess to what degree reconstructed systems agree in terms of attractor geometry, we require a mapping between the two state spaces that does not introduce any additional degrees of freedom, but consists only of 1) a projection into a space of the same dimensionality (and re-standardization of variables) followed by 2) a (geometry-preserving) rotation. This was to ensure that the quality of geometrical agreement can be attributed solely to the reconstruction method and not to any post-hoc fitting. For the first step, we used Principal Component Analysis (PCA) to reduce the dendPLRNN's latent space to the same dimensionality N as that of the ground truth system (which usually is of lower dimensionality). Afterwards, all axes were re-standardized (as for the original system). In a second step, a rotation matrix was then determined to rotate the latent state space such as to minimize the same Kullback-Leibler measure D_{stsp} that was used to assess agreement in attractor geometries, see Fig. A1 for an example. The optimal rotation matrix was determined by grid search over the space of rotation matrices, as we found numerical optimization to often yield inferior results. Note that this operation *does not alter the geometry* of objects in the latent space but merely rotates them such that they are best aligned with their ground truth counterparts (we also attempted Procrustes analysis (Gower, 1975), using the Procrustes Python library (Meng et al., 2022), to determine the best affine mapping between spaces directly, but found this generally to be inferior despite actually being less conservative than our approach). Comparing different grid- and step sizes in preliminary runs, we fixed parameters such that a single grid search takes no more than $30 - 60$ seconds on a single CPU. To confirm that this procedure yields results in agreement with those obtained from a co-trained linear-Gaussian model fed with continuous observations, we compared D_{stsp} computed in observation space ('D_{bin}') as outlined above to D_{stsp} computed on the corresponding dimensionality-reduced latent states using our PCA+rotation method ('D_{PCA}'). As shown in Fig. A1, these two measures were indeed highly correlated, $r \approx 0.94$. Example reconstructions from solely ordinal observations are given in Figs. 4 and A8. For the 6-dimensional Lewis-Glass chaotic network model, performing a grid search over rotation matrices in the observation space was unfortunately no longer computationally feasible, such that in this case we resorted to Procrustes analysis (see above; generally, the Procrustes method aims to superimpose two data sets by optimally translating, rotating, and scaling them, preserving geometric similarity). In this case, the correlation between the D_{stsp} measures obtained by a co-trained linear model and the one obtained post-hoc via the Procrustes-transformed space dropped to $r \approx 0.57$, but was still significant ($t_{28} = 3.77, p < .001$).

Power spectrum Hellinger distance The power spectrum Hellinger distance (D_H) was obtained by first sampling a time series of $100,000$ time steps and computing dimension-wise Fast Fourier Transforms (using `scipy.fft`) for both the ground truth system and simulated time series, after discarding transients. The noise-dominated high-frequency tails of the spectra were cut off, and the power spectra were slightly smoothed with a Gaussian kernel and normalized. We then computed the Hellinger distance (Mikhaeil et al., 2022) between smoothed power spectra of ground-truth, $F(\omega)$, and generated, $G(\omega)$, trajectories by

$$H(F(\omega), G(\omega)) = \sqrt{1 - \int_{-\infty}^{\infty} \sqrt{F(\omega)G(\omega)}d\omega} \in [0, 1] \tag{26}$$

The dimension-wise Hellinger distances were then averaged to yield the D_H values reported.

Spearman autocorrelation function (SACF) To assess temporal agreement between model-generated and ground truth time series for *ordinal* and *count* observations, we computed a measure based on the average SACF, defined as:

$$SACF_i(\tau) = \frac{\sum_{t=1}^{T-\tau} (r_{i,t} - \overline{r}_i) \cdot (r_{i,t+\tau} - \overline{r}_i)}{\sum_{t=1}^{T} (r_{i,t} - \overline{r}_i)^2}, \tag{27}$$

For this, we first sampled a time series of $100,000$ time steps and computed the dimension-wise Spearman autocorrelation for time lags up to 200 for both generated and ground truth test data (see Fig. A10). We then calculate the average squared error between the corresponding SACFs of ground truth and generated time series across all lags and dimensions as our performance metric:

$$\text{MSE}_{SACF} = \frac{1}{NT} \sum_{i=1}^{N} \sum_{\tau=1}^{T} \left(\text{SACF}_{\text{gen},i}(\tau) - \text{SACF}_{\text{ground truth},i}(\tau) \right)^2 \tag{28}$$

Mean squared prediction error A mean squared prediction error (PE) was computed across test sets of length $T = 10000$ by initializing the trained dendPLRNN with the test set time series up to some time point t from the estimated initial state $\mathbf{E}[\mathbf{z}_t|\mathbf{x}_{1:t}]$, from where it was then iterated forward by n time steps to yield a prediction at time step $t + n$. The n-step PE is then defined as the MSE between predicted and true observations:

$$PE(n) = \frac{1}{N(T-n)} \sum_{t=1}^{T-n} \sum_{i=1}^{N} (x_{i,t+n} - \hat{x}_{i,t+n})^2 \tag{29}$$

Due to exponential divergence of initially close trajectories in chaotic systems, the PE is sensible only for a limited number of time steps (Koppe et al., 2019b); here we chose $n = 10$.

Determining initial states for computing mean squared prediction errors Given exponential trajectory divergence, the prediction error (PE) also depends on the precise initialization of the DS model at time t. For SVAE training, an estimate of the initial state is directly provided by the encoder model. For BPTT-TF, the initial state $\mathbf{z}_t$ can be inferred by a jointly trained linear mapping from $\mathbf{z}_t$ to $\mathbf{x}_t$ (Brenner et al., 2022). For MTF and GVAE-TF, if $K = M$, the initial state $\mathbf{E}[\mathbf{z}_t|\mathbf{x}_{1:t}]$ is also directly given by the encoder model. If $K < M$, a subset of $M - K$ latent states is drawn from a standard normal distribution. In this under-specified case, we used a warm-up phase of t_w time steps, where the system was initialized from the encoder at time $t - t_w$ and iterated forward, providing encoded states in the form of TF signals at every time step, which yields an initial state estimate $\mathbf{E}[\mathbf{z}_t|\mathbf{x}_{(t-t_w):t}]$. We determined $t_w = 20$ by grid-search, leading to on average best results across datasets. We found that including this warm-up phase significantly improved prediction performance, although it still led to slightly worse results than for the fully specified case, $K = M$. Finally, for MS we neither have an encoder model nor a TF signal, and hence a sensible estimate for the initial state is difficult to obtain.

Ordinal prediction error The ordinal PE was computed similarly as the mean squared PE above, but – as pointed out in Öğretir et al. (2022) – due to the non-metric nature of ordinal data taking the absolute (L_1) deviation between observed and predicted values is more sensible:

$$OPE(n) = \frac{1}{N(T-n)} \sum_{t=1}^{T-n} \sum_{i=1}^{N} |o_{i,t+n} - \hat{o}_{i,t+n}| \tag{30}$$

Spearman cross-correlation (SCC) To assess whether the global cross-correlation structure between the different ordinal time series is preserved by the reconstruction method, the Spearman correlation between each pair of ordinal time series was computed based on $100,000$ time steps long samples, using `scipy.stats.spearmanr`, for both generated and ground truth test data.

$$\text{SCC}_{ij} = \frac{1}{T} \sum_{t=1}^{T} \left(\frac{r_{it} - \overline{r}_i}{\sigma_{r_i}} \right) \left(\frac{r_{jt} - \overline{r}_j}{\sigma_{r_j}} \right) \tag{31}$$

with σ_{r_i} and σ_{r_j} the standard deviation of the ranks of time series i and j. We then calculated the MSE across all elements of the respective correlation matrices for the N ordinal time series:

$$\text{MSE}_{SCC} = \frac{1}{N^2 - N} \sum_{i=1}^{N} \sum_{\substack{j \neq i}}^{N} \left(\text{SCC}_{\text{gen},ij} - \text{SCC}_{\text{ground truth},ij} \right)^2, \tag{32}$$

Lyapunov exponent The maximum Lyapunov exponent quantifies the divergence rate of nearby trajectories, and (for discrete-time systems $F_{\boldsymbol{\theta}}(\boldsymbol{z}_{t-1}, \boldsymbol{s}_t)$ like the dendPLRNN, Eq. 1) is defined as

$$\lambda_{max} := \lim_{T \to \infty} \frac{1}{T} \log \left\| \prod_{r=0}^{T-2} \boldsymbol{J}_{T-r} \right\|_2, \tag{33}$$

where $\boldsymbol{J}_t := \frac{\partial \boldsymbol{F}_{\boldsymbol{\theta}}(\boldsymbol{z}_{t-1}, \boldsymbol{s}_t)}{\partial \boldsymbol{z}_{t-1}} = \frac{\partial \boldsymbol{z}_t}{\partial \boldsymbol{z}_{t-1}}$ are the system's Jacobians and $\|\cdot\|_2$ is the spectral norm. To numerically approximate the Lyapunov spectrum for trained models, dendPLRNNs were iterated forward by 5500 time steps using Eq. 1, of which the first 500 steps were discarded to remove transients. For numerical stability, an algorithm based on Wolf et al. (1985); Vogt et al. (2022) was used which re-orthogonalizes the product series of Jacobians after every 5-th time step using a QR decomposition. For consistency, we also computed maximum Lyapunov exponents for the Lorenz ($\lambda_{max} = 0.903$), Rössler ($\lambda_{max} = 0.071$), and Lewis-Glass ($\lambda_{max} = 0.072$) models ourselves using the Julia library `DynamicalSystems.jl` (Datseris, 2018) and the dysts Python package (Gilpin, 2022), which both use the same algorithm by Wolf et al. (1985), but our estimates agree closely with those that can be found in the literature (Lorenz: $\lambda_{max} = 0.905$, Rössler: $\lambda_{max} = 0.072$, in Alligood et al. (1996))
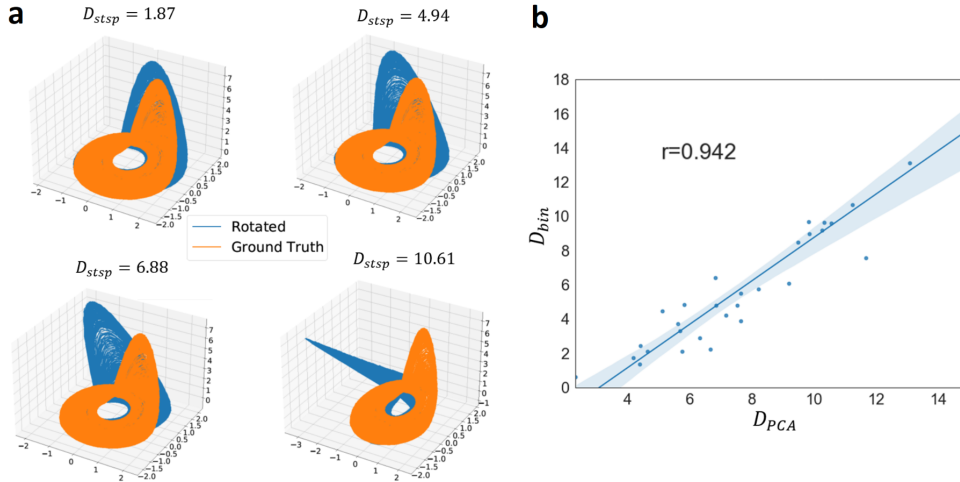

Figure A1. **a**: Ground truth and rotated attractors of the Rössler system with associated D_{stsp}-values. **b**: Correlation between geometrical reconstruction measures for the Rössler system directly in observation space given a co-trained linear (Gaussian) observation model (D_{bin}), and from a 3d PCA projection of latent space followed by an optimal rotation of the reconstructed attractor (D_{PCA}), based on a total of 30 trained models.

A.3. Comparison methods

As noted further above, different choices of DSR model are possible within our framework (see Appx. Table B3). Since the focus here is on evaluating the training framework for DSR from multimodal data itself, for comparability we used the same

dendPLRNN model in all five other methods compared to. The total number of trainable parameters was kept approximately the same for all methods (see also Table B1).

Multimodal SVAE First, we compared our method to a multimodal SVAE as proposed in (Kramer et al., 2022), optimizing its multimodal ELBO. This, to our knowledge, is currently the only other general approach specifically designed for DSR from arbitrary data modalities observed simultaneously. We followed the implementation of the encoder model as provided on `https://github.com/DurstewitzLab/mmPLRNN`, training with a sequence length of 150 time steps per batch, a hidden dimension of 20, and the parameters of the dendPLRNN the same as for the runs with MTF.

Classical RNN training with truncated BPTT Second, we tried 'classical' RNN training via truncated BPTT, where observations are provided as inputs at every time step, and the whole model is optimized using modality-specific decoder models, Eq. 22. Specifically, given observed multimodal time series $\boldsymbol{Y}$ of length T, observations $\boldsymbol{y}_t$ are included into the RNN updates according to

$$\boldsymbol{z}_t = \boldsymbol{A}\boldsymbol{z}_{t-1} + \boldsymbol{W}\phi(\boldsymbol{z}_{t-1}) + \boldsymbol{U}\boldsymbol{y}_t + \boldsymbol{h} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}) \tag{34}$$

where $\boldsymbol{U}$ is the input-to-hidden weight matrix. Training is performed on short subsequences of length T_{seq}, where T_{seq} is a hyper-parameter. The network then produces observations at each time step using the modality-specific decoder models, just as in MTF. During inference, the network is iterated forward by using the predicted observations at time t as input for the predictions at time $t + 1$. We probed different values for the sequence length $T_{seq} \in \{5 \ldots 50\}$. We found best settings to agree with the TF interval τ for MTF and the sequence length L for multiple shooting, but – consistent with previous work (Mikhaeil et al., 2022) – plain error gradient/ sequence truncation often led to divergence and failed to capture the true time scales in the fMRI data.

Multiple shooting A third approach that can deal with multimodal observation models but, unlike TF, does not require model inversion, is 'multiple shooting (MS)' (Bock & Plitt, 1984), a method suggested in the dynamical systems literature (Voss et al., 2004). Multiple shooting aims to solve boundary value problems by dividing time into sub-intervals, treating each sub-interval as a separate initial value problem, and then imposing continuity conditions between intervals. Multiple shooting has also been applied to DS reconstruction and Neural-ODE training (Iakovlev et al., 2022), where the initial conditions ('shooting nodes') for each interval are model parameters and continuity across intervals is enforced through a penalty term in the loss (Voss et al., 2004). Hence, rather than controlling trajectory flows through a sparse TF signal applied after forcing intervals τ, alternatively one may reset the latent model trajectory to an inferred initial condition after τ time steps. The advantage is that this method does not require the inversion of decoder models and is hence naturally suited to handle different data modalities without further care (i.e., retaining the distributional properties of the original data). More specifically, the observed time series $\boldsymbol{Y}$ is partitioned into N_{seq} subsequences $\boldsymbol{Y}^s, s = 1 \ldots N_{seq}$, of length L, and for each subsequence a new initial condition $\boldsymbol{\mu}_0^s$ is learned. During training, trajectories are freely generated for L time steps from $\boldsymbol{\mu}_0^s$ for each subsequence, and likelihoods for the observed trajectories $\boldsymbol{Y}^s$ are computed using the observation models from Eqs. 2-4. A consistency (penalty) term in the loss ensures continuity between subsequences according to

$$\mathcal{L}_{\text{MS}} = \lambda_{\text{MS}} \sum_{s=1}^{N_{seq}-1} ||F_\theta(\boldsymbol{z}_L^s) - \boldsymbol{\mu}_0^{s+1}||_2^2 \tag{35}$$

where F_θ in our case is the dendPLRNN, Eq. 1, λ_{MS} is a regularization parameter, and $F_\theta(\boldsymbol{z}_L^s) = F_\theta(F_\theta(\ldots F_\theta(\boldsymbol{\mu}_0^s))) = F_\theta^L(\boldsymbol{\mu}_0^s)$. The sequence length L plays a similar role as the teacher forcing interval τ for MTF, controlling the times at which states and gradients are reset during training. Indeed, best performance is achieved when using choosing $L = \tau$ for all datasets studied here (see Table B1).

Standard unimodal approach with data 'Gaussianization' A naive approach for handling multimodal observations with any type of DSR model would be to pre-process all modalities such as to bring them into approximate agreement with Gaussian assumptions. Thus, for training the dendPLRNN with standard BPTT-TF (Brenner et al., 2022) and GVAE-TF, we transformed ordinal and count observations into approximately Gaussian variables through a Box-Cox-transformation (Box & Cox, 1964), z-scoring, and Gaussian kernel smoothing across the time series. For the optimal width of the Gaussian kernel, we performed a grid search over kernel sizes $\nu \in \{0, 0.01, 0.1, 1, 10, 15, 20, 25\}$. Optimal settings for the results reported are given in Table B1.

A.4. Alternative encoder models

Mixture-of-Experts CNN encoder   The Mixture-of-Experts (MoE) approach (Shi et al., 2019) employs the same architecture as the CNN encoder described in Methods A.1, but uses a distinct encoder for each modality. We combined the outputs of each encoder into a joint estimate using a weighted average:

$$\boldsymbol{\mu}_{\text{MoE}} = w_g \boldsymbol{\mu}_g + w_o \boldsymbol{\mu}_o + w_c \boldsymbol{\mu}_c$$
$$\boldsymbol{\Sigma}_{\text{MoE}} = w_g \boldsymbol{\Sigma}_g + w_o \boldsymbol{\Sigma}_o + w_c \boldsymbol{\Sigma}_c,$$

with means $\boldsymbol{\mu}_{\{g,o,c\}}$, diagonal covariances $\boldsymbol{\Sigma}_{\{g,o,c\}}$ and weights w_g (Gaussian), w_o (ordinal) and w_c (count process), all set to $1/3$. We also tested a product-of-experts (PoE) approach (Hinton, 2002; Wu & Goodman, 2018), where the estimates of the individual experts are multiplied instead of summed, which however often led to numerical instabilities during training on discrete variables.

RNN encoder   For the RNN encoder (Cho et al., 2014), the hidden states $\boldsymbol{h}_t$ of an RNN, where observations $\boldsymbol{y}_t$ are provided as inputs at every time step (similar to Eq. 34), are used to map onto the parameters of the approximate posterior at every time step. Here we use the standard RNN implementation in `torch.nn.RNN`, where

$$\boldsymbol{h}_t = \tanh(\boldsymbol{W}\boldsymbol{h}_{t-1} + \boldsymbol{U}\boldsymbol{y}_t + \boldsymbol{b})$$
$$\boldsymbol{\mu}_t = \boldsymbol{W}_\mu \boldsymbol{h}_t + \boldsymbol{b}_\mu$$
$$diag\left([\log \sigma_1^2, \ \ldots, \ \log \sigma_K^2]\right) = \boldsymbol{W}_\Sigma \boldsymbol{h}_t + \boldsymbol{b}_\Sigma$$

with RNN parameters $\{\boldsymbol{W}, \boldsymbol{U}, \boldsymbol{b}\}$ and linear readout weights $\{\boldsymbol{W}_\mu, \boldsymbol{b}_\mu\}$ for the mean and $\{\boldsymbol{W}_\Sigma, \boldsymbol{b}_\Sigma\}$ for the logarithm of the diagonal covariance of the approximate posterior, respectively.

Transformer   The Transformer encoder is based on the architecture in (Vaswani et al., 2017). Since computation time scales with T_{seq}^2 in this architecture, we restricted the sequence length to 100 steps. We used positional encodings to acknowledge the sequential (time series) nature of the data, as proposed in (Vaswani et al., 2017). The sequence plus encodings is then passed through a standard Transformer encoder block using `torch.nn.TransformerEncoder`. As for the other encoder models, the output of the Transformer was mapped via linear readout layers onto the mean and logarithm of the covariance of the approximate posterior.

Multi-layer-perceptron (MLP)   We also tested a standard MLP with 3 layers and rectified linear unit (ReLU) nonlinearity as an encoder model, with outputs mapped via a linear readout layer to the mean and logarithm of the covariance of the approximate posterior, as for the other encoders tested.

Performance comparisons of these different encoder architectures are provided in Table B2.

A.5. Datasets

Lorenz-63 system   A stochastic version of the 3d Lorenz-63 system, originally proposed in Lorenz (1963), is defined by

$$
\begin{aligned}
\mathrm{d}x &= (\sigma(y - x))\mathrm{d}t + \mathrm{d}\epsilon_1(t), \\
\mathrm{d}y &= (x(\rho - z) - y)\mathrm{d}t + \mathrm{d}\epsilon_2(t), \\
\mathrm{d}z &= (xy - \beta z)\mathrm{d}t + \mathrm{d}\epsilon_3(t).
\end{aligned}
\tag{36}
$$

Parameters used for producing ground truth data in the chaotic regime were $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$. Process noise was injected into the system by drawing from a Gaussian $\mathrm{d}\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, 0.01^2 \mathrm{d}t \times \boldsymbol{I})$. For both training and test data, a trajectory of $100,000$ time steps was sampled, performing numerical integration with `scipy.odeint` ($\mathrm{d}t = 0.05$; note this value differs from the one used in Mikhaeil et al. (2022), explaining why different τ values were required for optimal reconstructions). To obtain multimodal observations, trajectories drawn from the ground truth system were fed into the different types of observation models in Eqs. 2-4, with randomly drawn parameters.

Symbolic representation of Lorenz-63 dynamics To obtain a symbolic representation of the Lorenz-63 system, we first sampled a time series as above, using Eq. 36, and removed transients. We then divided the system's state space into N^3 cubes (with $N = 4$ bins per dimension). The range on each axis covered by the bins was set to the minimum-to-maximum extent of the attractor. Each data point of the time series was then assigned a label corresponding to the cube it fell in, using a one-hot encoding of length N^3. In fact, as for the Lorenz-63 chaotic attractor 28 of the 64 bins turned out empty (did not contain any points of the time series), the number of symbols (i.e., vector length) could be reduced further to just 36 (see also Fig. 4 for decoded bin probabilities). The procedure is sketched in the bottom half of Fig. 4. We then furthermore performed a delay-embedding of the symbolic time series using a delay $\tau = 10$ and an embedding dimension of $d = 2$, thus doubling the length of the symbolic vector ($d = 3$ gave similar results; see Matilla-García et al. (2021) for a discussion of how to determine optimal delay embedding parameters for symbolic sequences, although not used for the present purposes). Although not strictly necessary, this step profoundly improved DS reconstruction from symbolic sequences.

Rössler system The Rössler system was introduced in Rössler (1976) as a simplified version of the Lorenz system, and is given (in SDE form) by

$$
\begin{aligned}
\mathrm{d}x &= (-y - z)\mathrm{d}t + \mathrm{d}\epsilon_1(t), \\
\mathrm{d}y &= (x + ay)\mathrm{d}t + \mathrm{d}\epsilon_2(t), \\
\mathrm{d}z &= (b + z(x - c))\mathrm{d}t + \mathrm{d}\epsilon_3(t).
\end{aligned}
\tag{37}
$$

Parameters used for producing ground truth data in the chaotic regime were $a = 0.2, b = 0.2$, and $c = 5.7$. Process noise was added by drawing $\mathrm{d}\epsilon \sim \mathcal{N}(\mathbf{0}, 0.01^2 \mathrm{d}t \times \boldsymbol{I})$. Training and test data was sampled as described above for the Lorenz-63 system, using $\mathrm{d}t = 0.1$.

Lewis-Glass chaotic network model We simulate a 6-dimensional model of a neural network, originally introduced in Lewis & Glass (1992). Here the individual units of the network are endowed with a continuous gain function $G(x) = \frac{1+\tanh(-\alpha x)}{2}$, with the vector field given by

$$
\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \frac{-\boldsymbol{x}}{\tau} + G(\epsilon\, \boldsymbol{K}\boldsymbol{x}) - \beta.
\tag{38}
$$

To sample from this system in the chaotic regime, we used the `Hopfield` model implementation in the Python package `dysts.flows`, based on Gilpin (2022). Here, $\alpha = -1, \beta = 0.5, \epsilon = 10, \tau = 2.5$, and

$$
\boldsymbol{K} = \begin{bmatrix}
0 & -1 & 0 & 0 & -1 & -1 \\
0 & 0 & 0 & -1 & -1 & -1 \\
-1 & -1 & 0 & 0 & -1 & 0 \\
-1 & -1 & -1 & 0 & 0 & 0 \\
-1 & -1 & 0 & -1 & 0 & 0 \\
0 & -1 & -1 & -1 & 0 & 0
\end{bmatrix}
$$

We generated training and test data by `maketrajectory`, and down-sampled the generated data by a factor of 30. We sampled ordinal and count data in the same way as for the other datasets. Example time series and reconstructions are displayed in Fig. A12.

Human fMRI dataset This same dataset was used previously in the study by Kramer et al. (2022) on multimodal data integration for DS reconstruction, and is openly available at https://github.com/DurstewitzLab/mmPLRNN. In the study, 26 participants (of which 20 were selected for analysis, cf. Kramer et al. (2022)) were shown a series of images of different shapes (rectangles and triangles) while undergoing fMRI with a sampling rate of $1/3$ Hz. The subjects were asked to identify the type of shape presented in the current image or in the preceding one, respectively, under three different task conditions: a continuous delayed response 1-back task (CDRT), a continuous matching 1-back task (CMT), and a 0-back control choice reaction task (CRT). A resting condition and an instruction phase were also included. Each task condition was repeated five times, where the last repetition of all task stages was left out as a test set in our analyses (see Fig.

A2**b**). The brain activity of the participants as assessed via the BOLD signal projected onto the first principal component within each of 10 different brain regions in both hemispheres that were identified to be relevant for this task. More details on this study can be found in Koppe et al. (2014). Due to the short length of these time series, for the fMRI results from Table 2 we chose a sequence length of 72 time steps for testing and training.

Hippocampal multiple single-unit (MSU) and spatial position data For our second empirical example, we used electrophysiological recordings from the rodent hippocampus and spatial location data (Grosmark et al., 2016), publicly available at `https://crcns.org/data-sets/hc/hc-11/about-hc-11`. Specifically, the "sessInfo.mat" file from rat 'Achilles', which contains extracted spike times, was chosen for exemplification and further preprocessed using the script from (Zhou & Wei, 2020), provided at `https://github.com/zhd96/pi-vae/blob/main/code/rat_preprocess_data.py`, to obtain counts per time interval, using a binning width of 200 ms. For our analysis, we focused exclusively on the MAZE task and selected the 60 most active neurons, as many neurons had very low (statistically insufficient) activity levels. The position data was represented as a continuous 1d vector, with missing values imputed from neighboring points. Rats received water rewards at both end points of the track, which were provided to the model as scalar external signals s_t (cf. Eq. 1) set to 1 for 5 time bins around the time points at which the rat started to move away from the reward location (with trainable weights $U \in \mathbb{R}^{M \times 1}$ in Eq. 1).
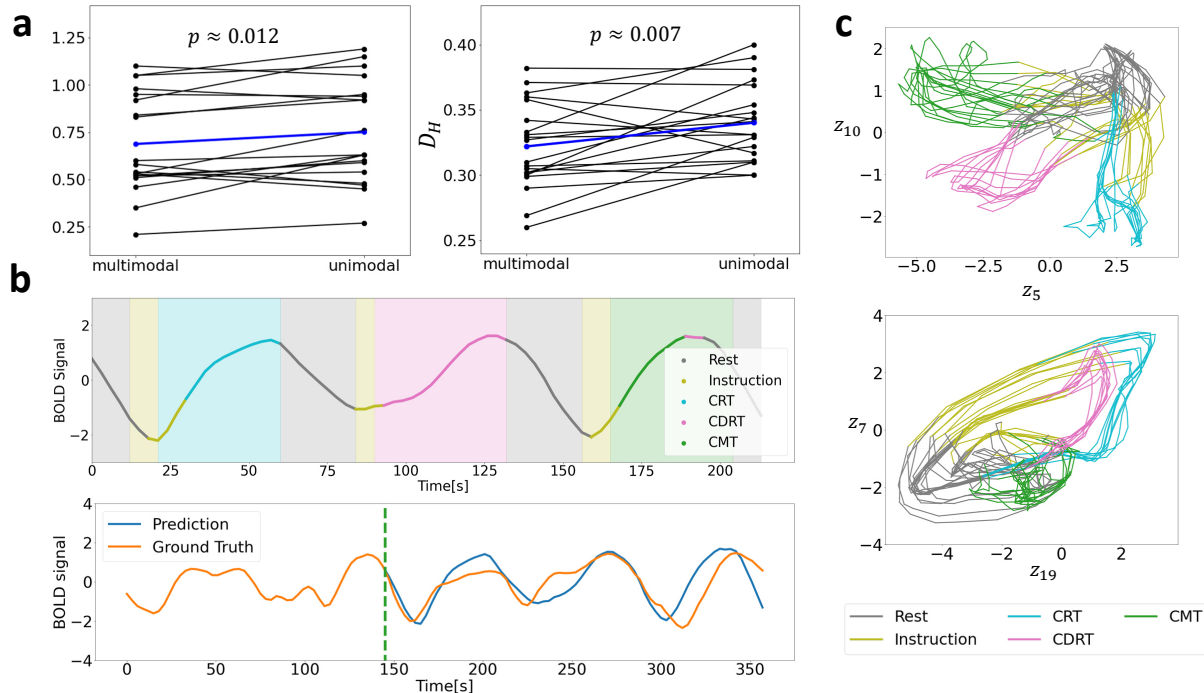
B. Supplemental Results

Figure A2. **a**: Multimodal integration using MTF on functional magnetic resonance imaging (fMRI)+behavioral data significantly helps to improve DS reconstruction compared to just fMRI data alone (unimodal). Results are for 20 subjects (black lines; blue line = mean), showing geometrical (D_{stsp}, left) and temporal (D_H, right) disagreement between true and reconstructed systems; p-values from paired t-tests as indicated. **b**: Example of decoded (color-coding of time series) and true (pale background colors) task stages for one brain area and subject. The trained model was freely iterated forward from the first time step of the test set, and task stages were decoded from the simulated activity. Decoding of class label $\hat{l} \in \{$Rest, Instruction, CRT, CDRT, CMT$\}$ is based on the maximum posterior probability, $\hat{l}_t = \arg\max p(l_{kt}|z_t)$, given the latent trajectory z_t. The graph below shows true and predicted BOLD signals. **c**: 2d subspaces of freely generated latent activity for a dendPLRNN ($M = 30, B = 10, K = 20, \tau = 7$) trained jointly on continuous and categorical data by MTF for one example subject. The color coding corresponds to the task labels predicted according to the maximum posterior probability given the latent state as in **b** at each time step. The latent space appears to be structured according to the different task stages.
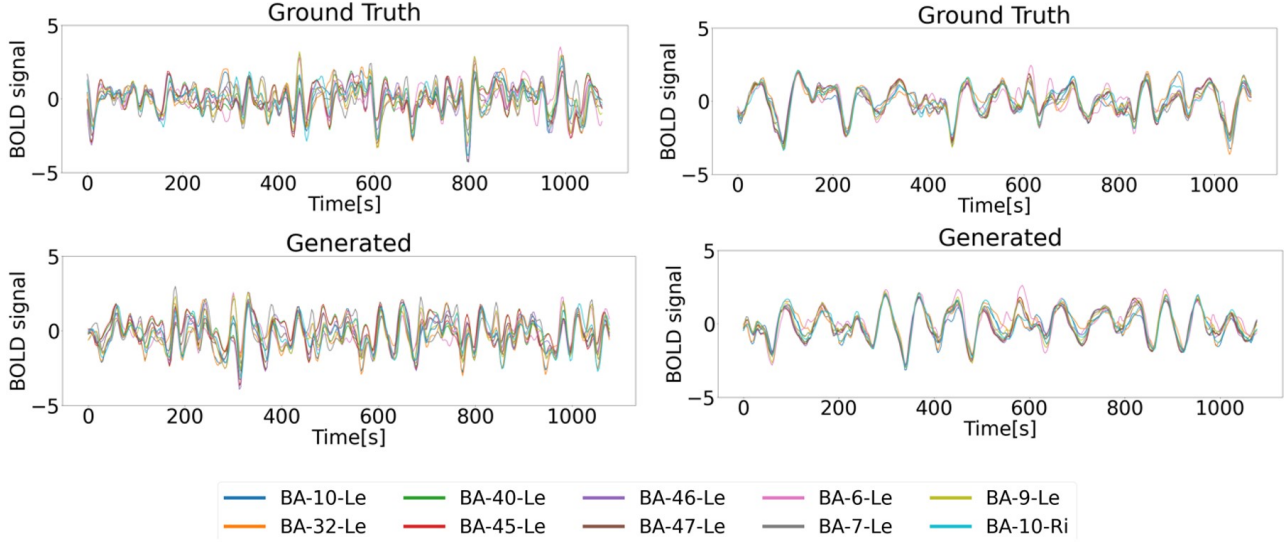
26

Figure A3. Freely generated time series from 10 brain areas per subject from subjects #3 (left) and #7. (right). The trained DSR model, only iterated by providing an initial state, captures the overall temporal structure of the complex activity patterns even from very short time series.
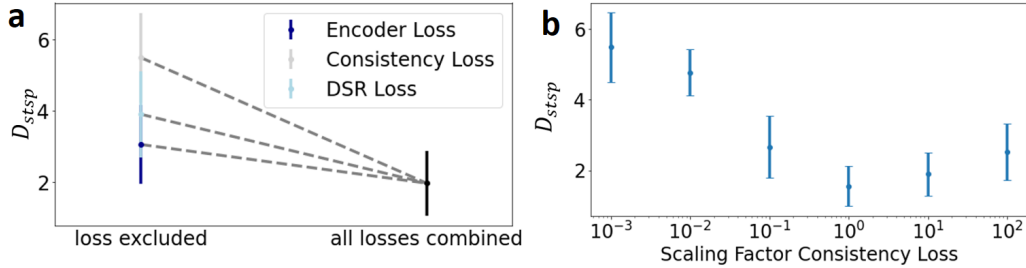

Figure A4. Impact of different loss terms in Eq. 11 on DS reconstruction quality, indicating that all loss components are crucially important. **a**: Comparison of state space agreement D_{stsp} when excluding different terms from the total loss in Eq. 11, and **b** for different scaling of the consistency loss ($\mathcal{L}_{con}$), for the dendPLRNN trained by MTF on data from the chaotic Lorenz system as in Sect. 3.1.

Table B1. Hyperparameter settings for MTF, SVAE, BPTT, GVAE-TF, and MS trained on the Lorenz, Rössler and Lewis-Glass model, and for the experimental fMRI data. For explanation of symbols, see Methods A.3.

Dataset	M	B	K	τ, L, T_{seq}	λ_{MS}	ν
Lorenz	20	15	15	10	1.0	10
Rössler	20	15	15	10	1.0	15
Lewis-Glass	20	15	15	20	1.0	20
fMRI	30	40	30	7	1.0	1

Table B2. Performance comparison of different encoder models used in the MTF framework, trained on multimodal data from the chaotic Lorenz system. PE is the prediction error and OPE the ordinal prediction error. Note that SCC (Spearman cross-correlation), OACF (ordinal autocorrelation function), and CACF (count autocorrelation function) all refer to mean-squared-errors (MSEs) between ground truth and generated correlation functions.

Encoder	$D_{stsp} \downarrow$	$D_H \downarrow$	PE $\downarrow$	OPE $\downarrow$	SCC $\downarrow$	OACF $\downarrow$	CACF $\downarrow$
CNN	**3.4 ± 0.35**	**0.30 ± 0.06**	**1.3e−2 ± 2e−4**	**0.12 ± 0.03**	**0.07 ± 0.01**	**0.07 ± 0.01**	**6.6e−5 ± 8e−6**
CNN-MoE	5.89 ± 0.18	0.43 ± 0.03	2.3e−2 ± 5e−4	0.13 ± 0.00	0.10 ± 0.00	0.19 ± 0.01	1.1e−4 ± 2e−5
RNN	5.47 ± 0.48	0.32 ± 0.04	1.6e−2 ± 2e−4	0.15 ± 0.01	0.13 ± 0.02	0.05 ± 0.01	8.5e−5 ± 9e−6
Transformer	5.85 ± 0.14	0.40 ± 0.04	4.8e−2 ± 5e−4	0.16 ± 0.00	0.17 ± 0.03	0.16 ± 0.02	9.5e−5 ± 7e−6
MLP	6.57 ± 0.14	0.43 ± 0.01	5.4e−2 ± 6e−4	0.15 ± 0.00	0.15 ± 0.01	0.21 ± 0.01	1.3e−4 ± 9e−6

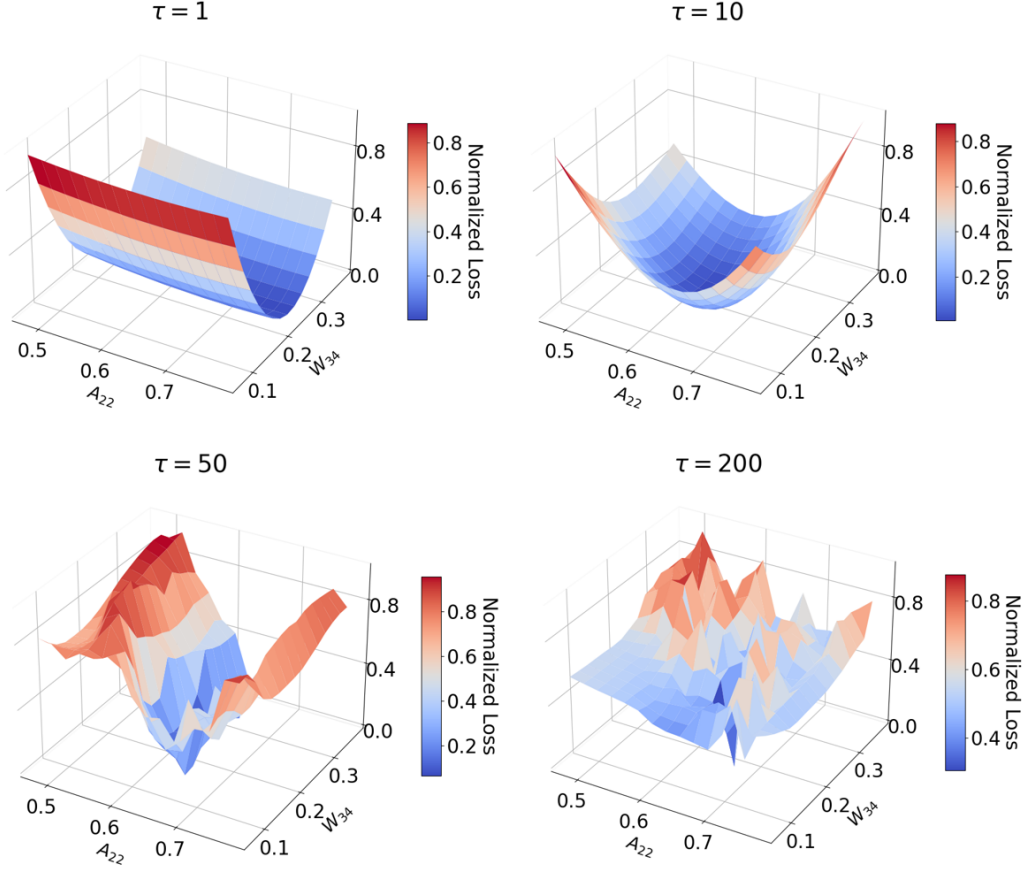*Figure A5.* MTF loss landscapes (Eq. 11) as a function of two randomly chosen parameters (A_{22}, W_{34}) of the DSR model (Eq. 1) for four different settings of the TF interval τ. The loss was computed on a random batch of training data for the model from Fig. 2. Smoother loss landscapes are obtained for lower τ (similar as observed for GTF in Hess et al. (2023)). While for the optimal TF interval $\tau = 10$ the loss is smooth and convex, for too low $\tau = 1$ flat directions in the loss appear.

Table B3. Performance comparison of different RNNs used as DSR model in the MTF framework, incl. the dendPLRNN (Brenner et al., 2022) used here, an LSTM (Hochreiter & Schmidhuber, 1997), and a GRU (Chung et al., 2014), all trained by MTF on multimodal data from the chaotic Lorenz system (10% Gaussian observation noise). PE is the prediction error and OPE the ordinal prediction error. Note that SCC (Spearman cross-correlation), OACF (ordinal autocorrelation function), and CACF (count autocorrelation function) all refer to mean-squared-errors (MSEs) between ground truth and generated correlation functions.

RNN Model	$D_{stsp} \downarrow$	$D_H \downarrow$	PE $\downarrow$	OPE $\downarrow$	SCC $\downarrow$	OACF $\downarrow$	CACF $\downarrow$
dendPLRNN	3.4 ± 0.35	0.30 ± 0.06	$1.3\mathrm{e}{-2} \pm 2\mathrm{e}{-4}$	0.12 ± 0.03	0.07 ± 0.01	0.07 ± 0.01	$6.6\mathrm{e}{-5} \pm 8\mathrm{e}{-6}$
LSTM	3.8 ± 0.74	0.31 ± 0.01	$5.4\mathrm{e}{-2} \pm 5\mathrm{e}{-4}$	0.16 ± 0.03	0.09 ± 0.02	0.09 ± 0.02	$8.8\mathrm{e}{-5} \pm 8\mathrm{e}{-6}$
GRU	3.47 ± 0.56	0.29 ± 0.03	$3.5\mathrm{e}{-2} \pm 5\mathrm{e}{-4}$	0.13 ± 0.03	0.06 ± 0.01	0.08 ± 0.01	$7.1\mathrm{e}{-5} \pm 5\mathrm{e}{-6}$
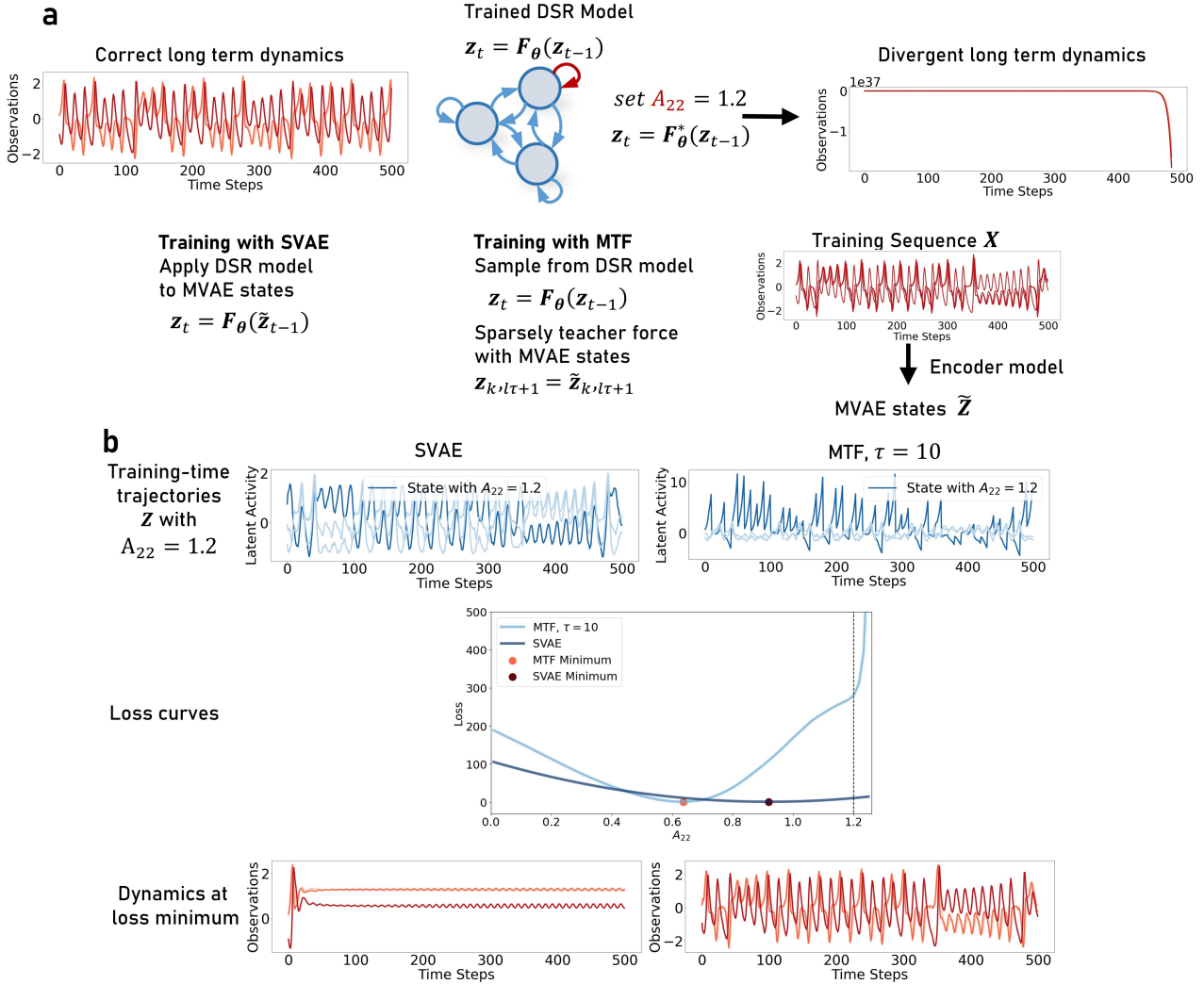
Figure A6. Illustration of how dramatic changes in long-term dynamics differently impact the SVAE and MTF loss. **a**: A dendPLRNN successfully trained on multimodal observations from the Lorenz-63 system (Fig. 2) is altered by setting a parameter of the linear self-connectivity $A_{22} > 1$, which results in globally diverging dynamics, yet locally still consistent with the Lorenz-63. **b**: The global divergence is clearly apparent in the MTF training-time trajectories $\mathbf{Z}$ generated using STF with interval $\tau = 10$ (right), within which the DSR model evolves freely, unlike the SVAE (left). This divergence leads to large increases in the MTF training loss (see MTF loss curve for $A_{22} > 1$), and hence is strongly penalized. This effect is essentially not present for the SVAE, where the global divergence induces no considerable effect on the training loss. Hence, the mismatch in global (long-term) dynamics remains unrecognized by the SVAE, explaining why the MTF approach is so superior. In fact, as shown at the bottom, at the minimum of the SVAE loss ($A_{22} \approx 0.966$), only short-term predictions are correctly reproduced (left), while the MTF at its minimum ($A_{22} \approx 0.637$) produces trajectories whose long-term dynamics agree in their temporal structure with those of the original Lorenz-63 (right).
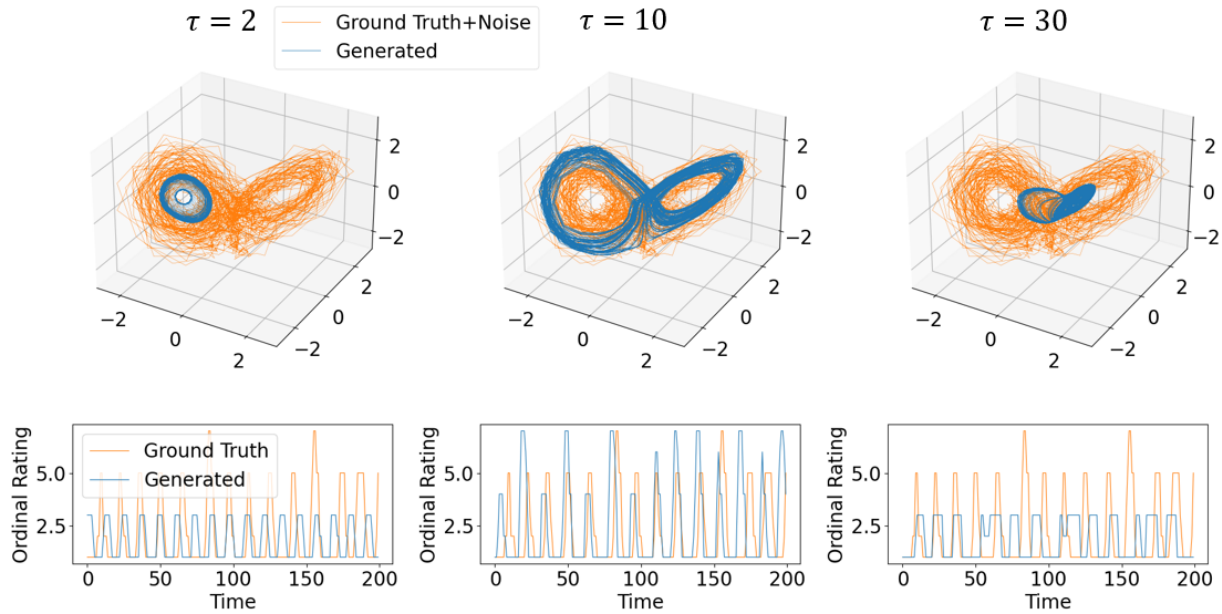
Figure A7. Example reconstructions (best out of 10 trained models each) of a dendPLRNN trained via MTF, compared to highly noisy Gaussian and ordinal training data (orange) produced by the Lorenz system. Reconstruction quality crucially depends on the choice of an optimal forcing interval, similar as observed for STF in Mikhaeil et al. (2022). Choosing the interval too small (left) or too large (right) leads to significantly worse reconstructions.
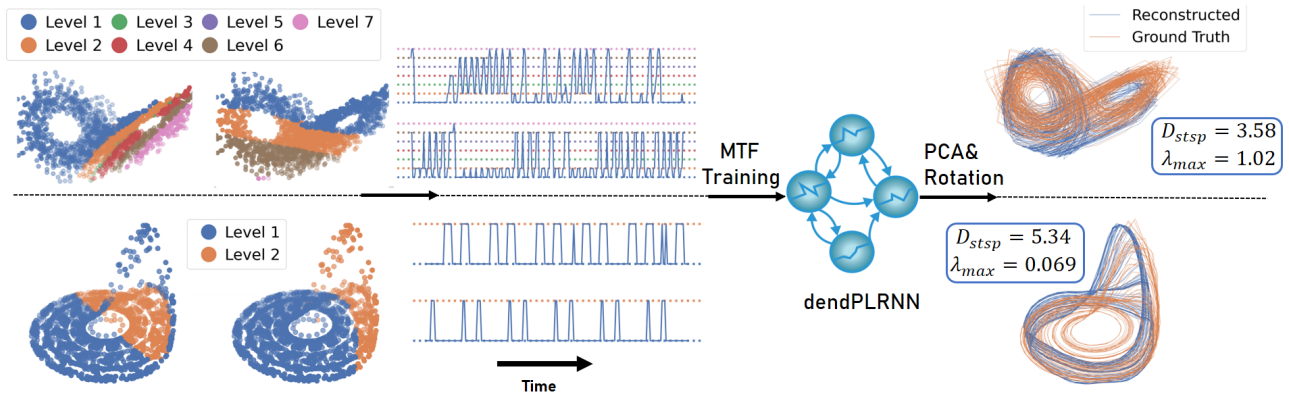

Figure A8. Top: DS reconstruction from eight ordinal observations with seven levels each produced by the Lorenz-63 system. Example traces of ordinal time series (center), and example reconstruction by MTF (right) $(M = 20, B = 10, K = 15, \tau = 10)$, preserving the butterfly geometry of the Lorenz attractor. Bottom: DS reconstruction of the Rössler attractor from from 75 ordinal observations with two levels each.
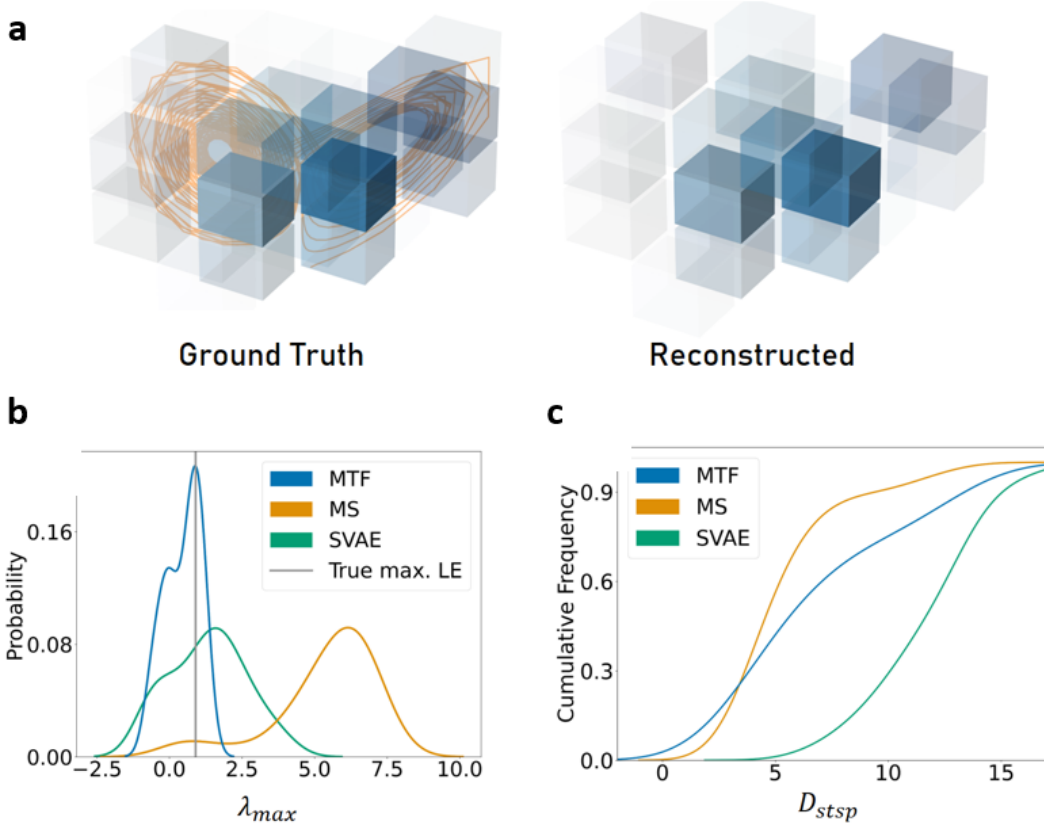
Figure A9. **a**: True and predicted class label probabilities (given the maximum posterior probability for a category at each time step) from a freely generated trajectory of a dendPLRNN ($M = 30, B = 15, K = 30, \tau = 10$), trained with MTF on the symbolic representation of the chaotic Lorenz-63 dynamics. **b** and **c**: Kernel-density estimates of distributions of maximum Lyapunov exponents (**b**) and cumulative distributions of D_{stsp} (**c**), comparing training with MTF, Multiple Shooting (MS), and multimodal SVAE across 30 trained models, all using the same dendPLRNN model. While for the MTF the estimates of Lyapunov exponents are centered around the true value ($\lambda_{max} = 0.903$), for SVAE and MS the distributions were much farther off.

Table B4. Comparison of dendPLRNN, eq. 1, trained by MTF (proposed method), by a sequential multimodal VAE (SVAE) based on (Kramer et al., 2022), and a multiple-shooting (MS) approach, on 8 ordinal observations with seven ordered categories, produced by the chaotic Lorenz system, Rössler system, and Lewis-Glass model, and on a symbolic representation of the chaotic Lorenz system. Values are mean ± SEM, averaged over 15 trained models. X = value cannot easily be computed for MS (because here initial conditions cannot be obtained directly from the data but require additional parameters).

Dataset	Method	$D_{stsp} \downarrow$	λ_{max}	OPE $\downarrow$	SCC $\downarrow$	OACF $\downarrow$
	MTF	8.8 ± 0.59	0.92 ± 0.39	0.24 ± 0.015	0.085 ± 0.02	0.016 ± 0.04
Lorenz-ordinal	SVAE	14.7 ± 0.7	0.44 ± 0.71	0.8 ± 0.03	0.17 ± 0.02	0.23 ± 0.02
	MS	13.8 ± 1.1	0.47 ± 0.67	X	0.24 ± 0.06	0.15 ± 0.03
	MTF	7.9 ± 0.8	0.03 ± 0.07	0.093 ± 0.007	0.051 ± 0.009	0.051 ± 0.009
Rössler-ordinal	SVAE	11.5 ± 1.3	-0.27 ± 0.58	0.39 ± 0.02	0.23 ± 0.05	0.18 ± 0.04
	MS	14.1 ± 1.0	-0.05 ± 0.12	X	0.12 ± 0.04	0.14 ± 0.03
	MTF	0.89 ± 0.04	-0.11 ± 0.41	0.15 ± 0.02	0.28 ± 0.05	0.15 ± 0.03
Lewis-Glass-ordinal	SVAE	1.40 ± 0.22	-1.8 ± 2.1	0.29 ± 0.01	0.49 ± 0.04	0.24 ± 0.02
	MS	1.0 ± 0.14	-0.14 ± 0.31	X	0.51 ± 0.04	0.45 ± 0.03
	MTF	4.4 ± 2.69	0.67 ± 0.37			
Lorenz-symbolic	SVAE	12.02 ± 1.98	1.87 ± 0.88			
	MS	4.46 ± 1.82	5.67 ± 1.25			

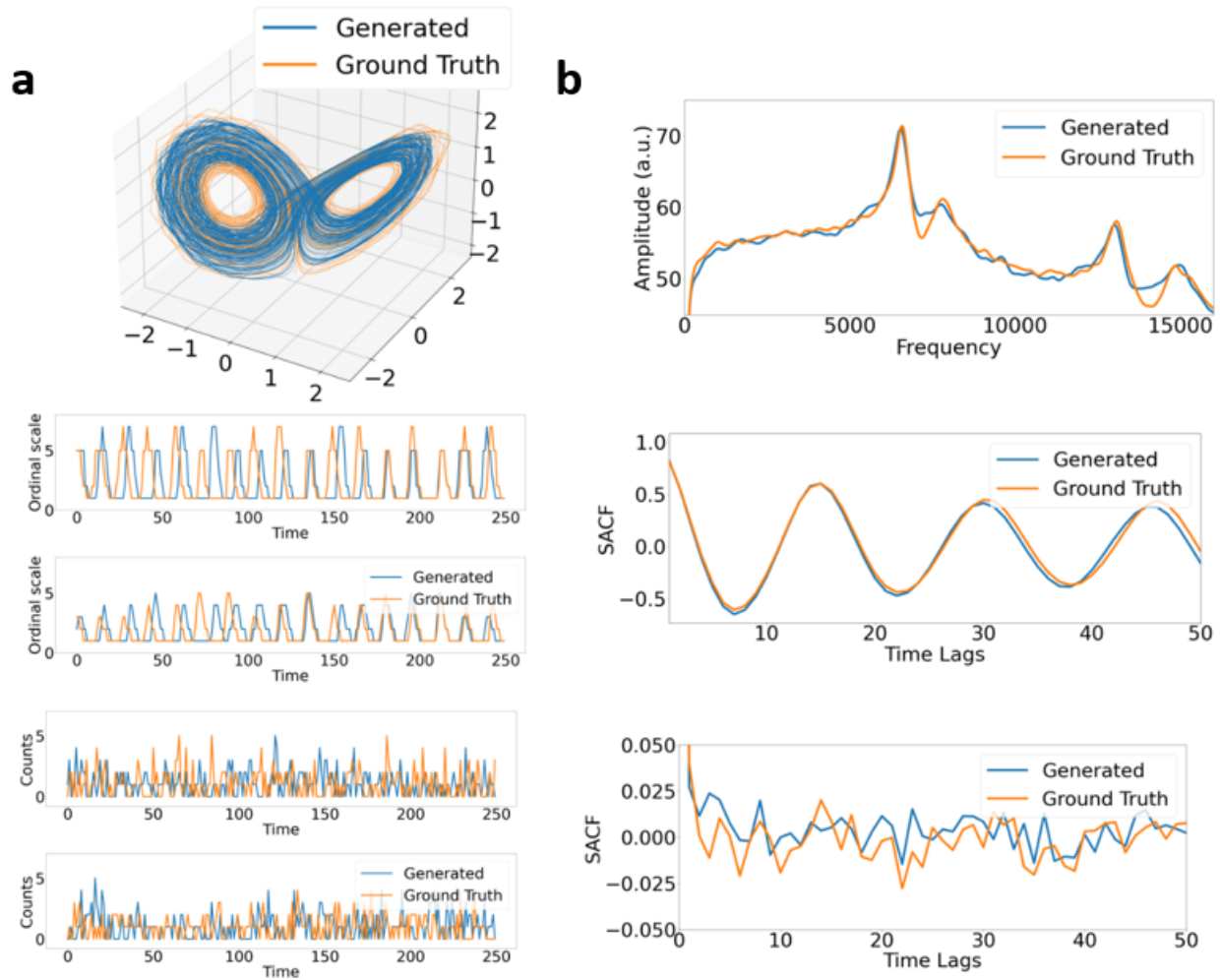*Figure A10.* **a**: Freely generated example trajectories and time series from a dendPLRNN ($M = 20,\ B = 10,\ K = 15,\ \tau = 10$) trained with MTF jointly on Gaussian, ordinal, and count data sampled from a Lorenz-63 system. **b**: Example power spectra (Gaussian data) and Spearman autocorrelation functions (ordinal and count data). Simulated latent trajectories faithfully capture the geometry of the Lorenz attractor, as well as the temporal structure of the ground truth data when projected back into observation space.
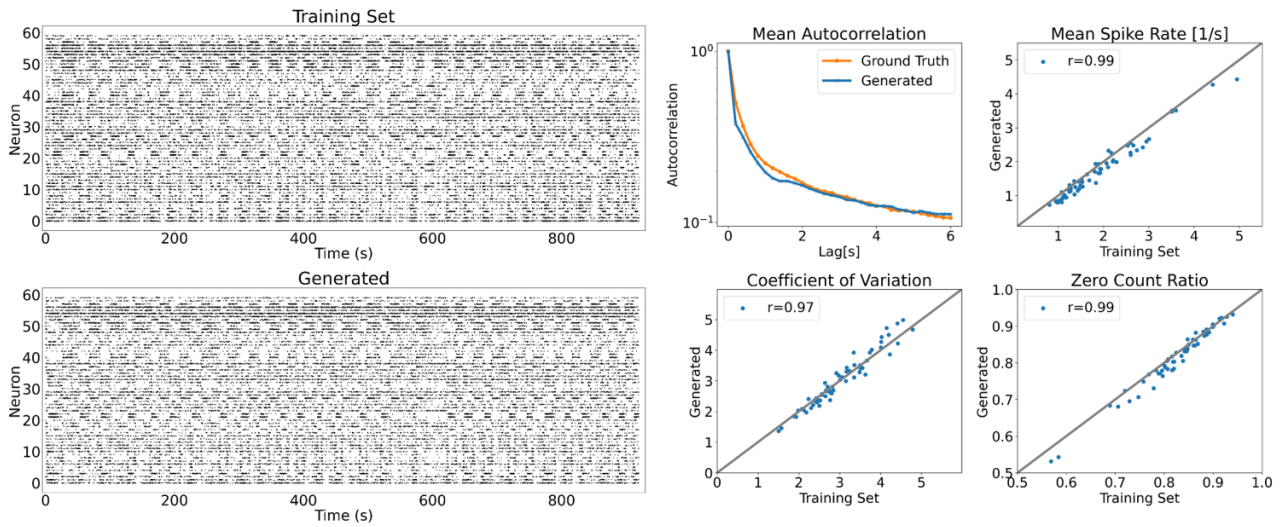
Figure A11. Example reconstructions of spike trains and spike statistics on the training set (see Methods A.5). Test set reconstructions and further statistics are in Figure 5.
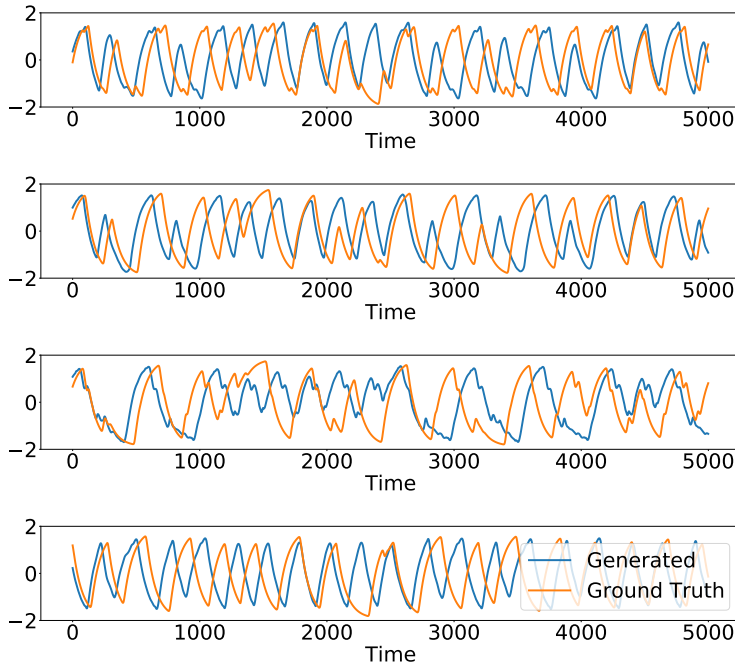

Figure A12. Example ground truth time series and freely generated series from a dendPLRNN ($M = 20, B = 10, K = 15, \tau = 20$) trained with MTF on the Lewis-Glass neural network model (Lewis & Glass, 1992).
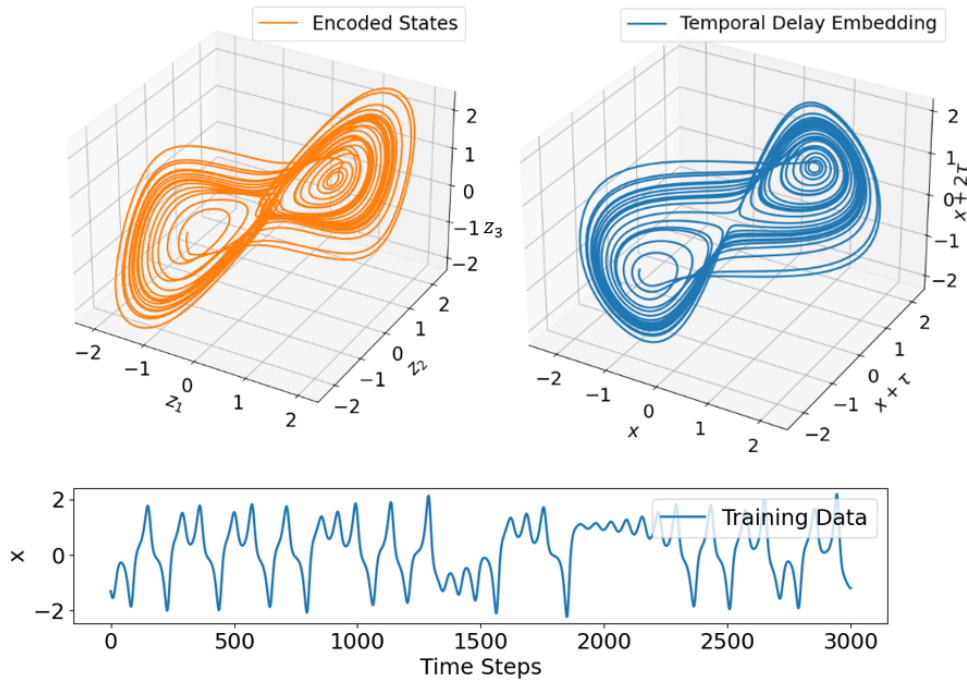
Figure A13. Reconstruction with missing variables (partial observations). Mean encoded states $\langle \tilde{Z} \rangle$ (left) after training a shPLRNN (Hess et al., 2023) with MTF on just one-dimensional time series from the Lorenz-63 system (x coordinate, bottom). Note that the reconstructed attractor's geometry closely resembles that of the true Lorenz-63 obtained from an optimal temporal delay embedding of the x variable (right; time lag chosen as the first minimum of the mutual information, see Kantz & Schreiber (2004)). This illustrates that MTF can perform a kind of implicit delay embedding when the underlying system is not fully observed.
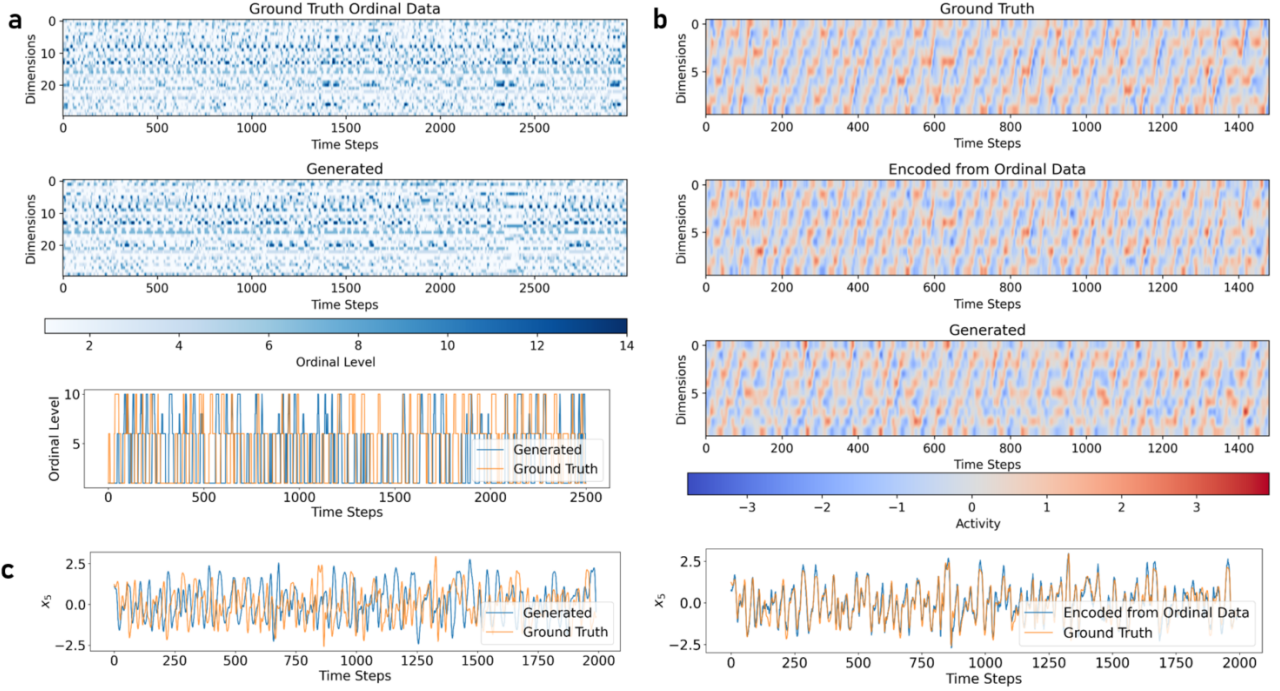
Figure A14. Reconstruction of a 10d chaotic Lorenz-96 system solely from ordinal observations with up to 15 levels using a shPLRNN (Hess et al. (2023); $M = 10, L = 100, \tau = 10$) as DSR model. On average, each variable only occupied 7 unique categories due to the random initialization of the observation model. **a**, top: Ground truth ordinal time series sampled from a randomly initialized ordinal observation model $p(\boldsymbol{o}_t|\boldsymbol{x}_t)$ from ground truth states $\boldsymbol{x}_t$ of the Lorenz-96 system, and reconstructed (generated) ordinal observations using the trained decoder model $p(\boldsymbol{o}_t|\tilde{\boldsymbol{z}}_t)$. Bottom: Example ground truth and freely generated ordinal time series from 1 channel. **b**: Ground truth states $\boldsymbol{x}_t$ (top), states encoded using the mean of the states taken from the trained MTF encoder $p(\tilde{\boldsymbol{z}}_t|\boldsymbol{o}_t)$ (center), and *freely generated* latent activity from the trained DSR model $\boldsymbol{z}_t = F_\theta(\boldsymbol{z}_{t-1})$ (bottom). A linear operator $\boldsymbol{B}$ was optimized to align ground truth states $\boldsymbol{X}$ of the Lorenz-96 (not seen during training!) and states $\tilde{\boldsymbol{Z}}$ encoded from the ordinal data via linear regression $(\boldsymbol{B} = \left(\tilde{\boldsymbol{Z}}^T\tilde{\boldsymbol{Z}}\right)^{-1}\tilde{\boldsymbol{Z}}^T\boldsymbol{X})$, and used to project the freely generated activity of the shPLRNN into the observation space of the Lorenz-96 system. **c**, left: Example of ground truth (orange) and freely generated (blue) activity, bearing the same temporal structure. Right: Aligned ground truth ($\boldsymbol{x}_t$) and encoded latent states ($\tilde{\boldsymbol{z}}_t$) as in b for one example unit. Note that the ground truth states $\boldsymbol{x}_t$ have *never been seen by the model during training* but are only inferred from the ordinal observations via the MTF encoder $p(\tilde{\boldsymbol{z}}_t|\boldsymbol{o}_t)$, yet still overlap almost perfectly.