
Reconstructing Nonlinear Dynamical Systems from Multi-Modal Time Series

Daniel Kramer,^{1*} Philine Lou Bommer,^{2*} Carlo Tombolini,¹ Georgia Koppe^{1,3} and Daniel Durstewitz^{1,4}

Abstract

Empirically observed time series in physics, biology, or medicine, are commonly generated by some underlying dynamical system (DS) which is the target of scientific interest. There is an increasing interest to harvest machine learning methods to reconstruct this latent DS in a data-driven, unsupervised way. In many areas of science it is common to sample time series observations from many data modalities simultaneously, e.g. electrophysiological and behavioral time series in a typical neuroscience experiment. However, current machine learning tools for reconstructing DSs usually focus on just one data modality. Here we propose a general framework for multi-modal data integration for the purpose of nonlinear DS reconstruction and the analysis of cross-modal relations. This framework is based on dynamically interpretable recurrent neural networks as general approximators of nonlinear DSs, coupled to sets of modality-specific decoder models from the class of generalized linear models. Both an expectation-maximization and a variational inference algorithm for model training are advanced and compared. We show on nonlinear DS benchmarks that our algorithms can efficiently compensate for too noisy or missing information in one data channel by exploiting other channels, and demonstrate on experimental neuroscience data how the algorithm learns to link different data domains to the underlying dynamics.

1. Introduction

Many natural phenomena, from physics to psychology, as well as many engineered systems, can genuinely be described as (usually nonlinear) dynamical systems (DSs), whose temporal evolution is specified by a set of differential or time-recursive equations. While traditionally these systems are derived by scientific insight, in recent years there has been growing interest to infer the governing equations directly from time series observations, in a purely data-driven way, using machine learning tools, such as polynomial regression (Brunton et al., 2016; Champion et al., 2019), Gaussian processes (Duncker et al., 2019), or recurrent neural networks (RNNs) (Lu et al., 2017a; Durstewitz, 2017b; Koppe et al., 2019; Hernandez et al., 2018; Vlachas et al., 2018; Pathak et al., 2018). Based on Cybenko’s universal approximation theorem (Cybenko, 1989), it has been shown that RNNs with sigmoid (Funahashi & Nakamura, 1993; Kimura & Nakano, 1998; Hanson & Raginsky, 2020) or Rectified Linear Unit (ReLU) (Lu et al., 2017b; Lin & Jegelka, 2018) activation functions are theoretically powerful enough to approximate any DS, i.e. its vector field, to arbitrary precision in its own set of dynamical equations (Funahashi & Nakamura, 1993; Trischler & D’Eleuterio, 2016). The objective of reconstructing, or approximating, the underlying DS itself, is more challenging compared to the more common goal of training a system to produce good ahead-predictions of temporal sequences (Koppe et al., 2019). This is because in DS reconstruction we require the trained model to also reproduce *invariant* properties of the underlying system in the limit $t \rightarrow \infty$, including the geometrical structure of its limit sets (attractors) or its temporal structure as assessed by the power spectrum.

Many natural and engineered DSs can be observed through many different measurement channels that produce time series: In Smartphone apps tracking psychiatric risk and behavior, for instance, one may want to combine categorical or ordinal information from ecological momentary assessments (EMA) with different continuous sensor readings, typing dynamics, proxies for social interactions, and GPS tracking (Radu et al., 2016; Koppe et al., 2018). In typical experiments in neuroscience one observes at the same time both continuous, often high-dimensional, measurements from the brain by means of electrophysiological or neuroimaging tools, and a subject’s often categorical behavioral responses

*Equal contribution ¹Dept. of Theoretical Neuroscience, Central Institute of Mental Health, Heidelberg University, Germany. ²Dept. of Machine Learning, Technical University Berlin, Berlin, Germany. ³Clinic for Psychiatry and Psychotherapy, Central Institute of Mental Health, Mannheim, Germany. ⁴Faculty of Physics and Astronomy, Heidelberg University, Germany.. Correspondence to: Daniel Durstewitz <daniel.durstewitz@zi-mannheim.de>, Philine Lou Bommer <philine.l.bommer@tu-berlin.de>.

across many trials. Not only is it often desirable to directly relate these different data streams within a common latent model, e.g. to gain insight into how neural activity produces behavior, or to predict behavioral choices from accelerometer readings, but the different data streams may also convey complementary information about the underlying DS that will supplement each other and help in identifying the system. Yet, different types of time series data may require very different distributional assumptions, especially when dealing with both continuous or ordinal data and categorical, event-type information.

While in general the integration of multi-modal data into common predictive models has been intensely researched in recent years (Sui et al., 2012; Lahat et al., 2015; Purdon et al., 2010; Ngiam et al., 2011; Srivastava & Salakhutdinov, 2012; Turner et al., 2013; Liang et al., 2015; Dezfouli et al., 2018; Halpern et al., 2018; Bhagwat et al., 2018; Antelmi et al., 2018; Sutter et al., 2021; Shi et al., 2021), so far this has hardly been a topic in the field of DS reconstruction. The major aim of the present work is to contribute to filling this gap. We consider the reconstruction of latent nonlinear DSs from observed time series that come from qualitatively different data domains best described by different distributional models (Fig. 1). We discuss several such observation (or decoder) models from the class of generalized linear models (GLMs), but focus for most of our presentation on the case where we have both continuous Gaussian (like neural measurements) as well as distinct categorical (like behavioral information) time series, linked to the same latent RNN for approximating the DS. For training the complete system, both a novel expectation-maximization (EM) as well as a variational inference (VI) approach are developed.

2. Related Work

A larger body of work deals with the identification of DSs from time series data. Some of these build on physical or biological domain knowledge to set up a system of ordinary (ODE) or partial (PDE) differential equations whose parameters are to be inferred from data (Gorbach et al., 2017; Raissi, 2018; Meeds et al., 2019), i.e., with the basic form of the ODE/PDE equations assumed to be known. Here, in contrast, we are interested in the case where the (exact) form of the equations is not known in advance, or where the data-generating DS is so complex (like the brain) that not all its details can be modeled, and hence we must rely on general purpose equations to approximate the underlying DS. Techniques toward this goal have been formulated both in continuous (Chen et al., 2021; Iakovlev et al., 2021) and discrete time (Schmidt et al., 2021; Zhao & Park, 2020). When using continuous-time ODE or PDE formulations (Chen et al., 2018; 2021; Iakovlev et al., 2021), numerical integration techniques must be used (Press et al., 2007), which

can be computationally quite expensive if the ODE/PDE system is stiff (like in spiking neuron models) and simple Euler or Runge-Kutta integration schemes quickly run into numerical issues (Press et al., 2007; Koch & Sevèg, 2003). Existing methods either aim to approximate the estimated vector field (usually obtained by differencing the time series) through (deep) neural networks (Trischler & D’Eleuterio, 2016; Chen et al., 2018), RNNs (Vlachas et al., 2018), or other types of universal approximators like polynomial basis expansions (Brunton et al., 2016; Champion et al., 2019). Or they are directly trained on the observed time series (Koppe et al., 2019; Schmidt et al., 2021; Lu et al., 2017a; Razaghi & Paninski, 2019; Hernandez et al., 2018), thus avoiding computation of numerical derivatives which are often unstable with large variance (Chen et al., 2017; Raissi, 2018; Baydin et al., 2018).

Most of the existing techniques assume (implicitly or explicitly) the underlying set of equations to be deterministic, i.e. do not consider dynamical process noise. This is especially true for continuous-time approaches (Ayed et al., 2019; Champion et al., 2019; Rudy et al., 2019), since stochastic DEs are even harder to deal with (Risken, 1984; Hertag et al., 2014). Here instead, following (Durstewitz, 2017b; Koppe et al., 2019; Schmidt et al., 2021), we assume that the generating equations are stochastic, which also helps in compensating for model misspecification (Abarbanel, 2013). Fully probabilistic, generative models for DS inference have been proposed in the context of state space models and the EM algorithm (Roweis & Ghahramani, 2002; Yu et al., 2006; Durstewitz, 2017b; Koppe et al., 2019; Schmidt et al., 2021) or, more recently, based on sequential variational auto-encoders (SVAE) (Krishnan et al., 2015; Archer et al., 2015; Zhao & Park, 2020; Kusner et al., 2017; Hernández et al., 2018; Pandarinath et al., 2018). Many of these were primarily aimed, however, at obtaining a smoothed posterior estimate $p(\mathbf{Z} | \mathbf{X})$ of latent state trajectories $\mathbf{Z} = \{\mathbf{z}_t\}$ given observed time series $\mathbf{X} = \{\mathbf{x}_t\}$. In DS reconstruction, in contrast, we demand that the RNN, once trained, will produce *simulated* data (i.e., generated from scratch from the prior $p(\mathbf{Z})$) with the same temporal and geometrical structure as those from the original DS (Molano-Mazon et al., 2018; Schmidt et al., 2021). Once successful reconstruction of the underlying DS has been achieved, it can be analyzed further with regards to its dynamical and mechanistic properties (e.g., Brunton & Kutz 2019; Durstewitz 2017a; Strogatz 2018). For instance, computational theories in neuroscience are often cast in terms of DSs (Wilson, 1999), and their properties like multi-stability (Durstewitz et al., 2000), cycles (Pandarinath et al., 2018) or line attractors (Machens et al., 2005; Durstewitz, 2003) are supposed to implement functions like working memory, motor activity, decision making, or timing. Such properties and the underlying theories can be probed on DSs reconstructed

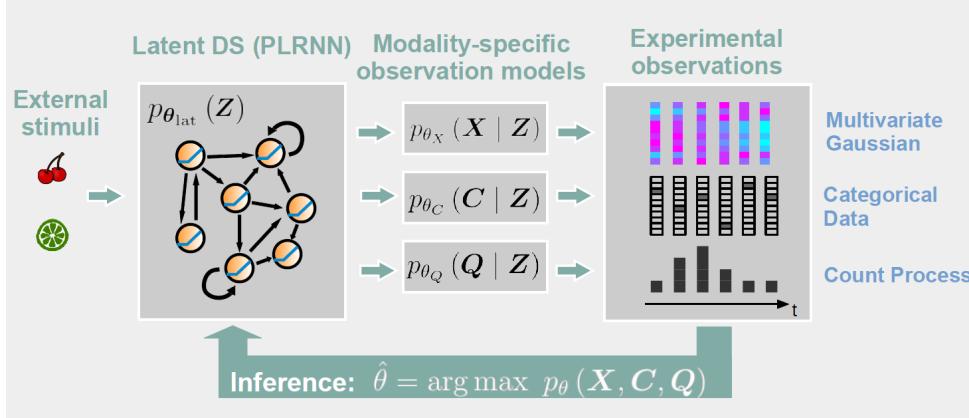


Figure 1: Illustration of the multi-modal PLRNN setup: A latent DS, modelled by a PLRNN that may potentially receive external inputs, is coupled to different data modalities via modality-specific observation models.

from empirical data.

Regarding integration of multi-modal data, especially in the fields of computer vision (Srivastava & Salakhutdinov, 2012; Sutter et al., 2021; Shi et al., 2021) and in medical AI applications (Purdon et al., 2010; Liang et al., 2015; Miotto et al., 2016; Rajkomar et al., 2018; Antelmi et al., 2018) it has been demonstrated that the fusion of different data domains into a common latent representation could substantially improve predictions or reveal interesting cross-domain links (Liang et al., 2015). For instance, integration of auditory and visual information into a common latent code improves speech recognition when the auditory signal is distorted, and enables cross-domain prediction (Ngiam et al., 2011; Lee et al., 2021). Likewise, integration of physiological measurements like electrocardiograms or blood pressure with (categorical) entries from electronic health records (EHR) does not only enable to find important links between different data types, but also leads to better prediction of clinical outcomes (Purdon et al., 2010; Liang et al., 2015; Rajkomar et al., 2018; Antelmi et al., 2018). Most recent work focused on variational auto-encoders (VAE) and inference for multi-modal integration, assuming fully joint (Vedantam et al., 2018), factorized (Kurle et al., 2019), mixture forms (Shi et al., 2019), or a combination of these (Sutter et al., 2021), for the approximate posterior. Comparatively less work on multi-modal VAEs has been done, however, in the time series domain, with some exceptions especially in the area of language processing (Tsai et al., 2019; Wu & Goodman, 2018).

Most importantly, none of the multi-modal approaches so far was aimed at *DS reconstruction* in the sense defined further above. Thus, to our best knowledge, algorithms for identifying nonlinear DSs from multiple diverse (esp. non-Gaussian) data modalities do not exist currently, although such scenarios frequently occur in the natural sciences. Here

we aim to fill this gap. We will also illustrate how this allows for new types of analysis by linking different data streams to the same latent DS.

3. Model Framework for DS Reconstruction from Multi-Modal Data

Our complete model setup is illustrated in Fig. 1. We assume that we have observed time series $\mathbf{X} = \{\mathbf{x}_t\}$ generated by some unknown DS $d\mathbf{y}/dt = f(\mathbf{y}, t)$ sampled at discrete time points t according to some output distribution $p(\mathbf{X} | \mathbf{Y})$, e.g. Gaussian, Poisson, or categorical. In fact, as illustrated in Fig. 1, here we assume that the unknown DS is observed through several such output channels (data modalities) simultaneously, from which we would like to infer the underlying DS which is approximated by a sufficiently expressive RNN. Here we assume that all observed time series were generated by the same underlying DS and hence are naturally aligned via their common time labels (reflecting the most common scenario in the natural sciences where measurements across different modalities are taken simultaneously).

3.1. Generative Multi-Modal RNN Model

For our approach we build on a nonlinear state space model framework introduced previously in (Durstewitz, 2017b; Koppe et al., 2019). The latent process used for approximating the unknown DS $f(\mathbf{y}, t)$ is modeled by a Gaussian piecewise-linear (PL) RNN of the form

$$\begin{aligned} \mathbf{z}_t | \mathbf{z}_{t-1} &\sim \mathcal{N}(\mathbf{A}\mathbf{z}_{t-1} + \mathbf{W}\phi(\mathbf{z}_{t-1}) + \mathbf{h} + \mathbf{F}\mathbf{s}_t, \Sigma), \\ \mathbf{z}_1 &\sim \mathcal{N}(\boldsymbol{\mu}_0 + \mathbf{F}\mathbf{s}_1, \Sigma), \end{aligned} \quad (1)$$

where $\mathbf{z}_t \in \mathbb{R}^{M \times 1}$ is the latent state vector, $\mathbf{A} \in \mathbb{R}^{M \times M}$ is diagonal with auto-regression weights a_{mm} , $\mathbf{W} \in \mathbb{R}^{M \times M}$ is off-diagonal (to minimize redundancy with terms in

A) with coupling weights w_{ml} , $m \neq l$, $\mathbf{h} \in \mathbb{R}^{M \times 1}$, and $\phi(\mathbf{z}_t) = \max(\mathbf{z}_t, 0)$ is an element-wise ReLU transform. We also account for a time-dependent external input $\mathbf{s}_t \in \mathbb{R}^{Q \times 1}$, weighted by coefficient matrix $\mathbf{F} \in \mathbb{R}^{M \times Q}$, as well as Gaussian process noise with diagonal covariance matrix $\Sigma \in \mathbb{R}^{M \times M}$. As argued in Durstewitz (2017b), this model may be seen as a discretization of a neural population model (although not important for the present purpose of DS reconstruction).

One major advantage of the specific PLRNN structure in the context of DS reconstruction is that many of its dynamical properties are (analytically) tractable: Fixed points and cycles of the system can be explicitly computed (Schmidt et al., 2021), many important types of bifurcations are comparatively well described for this class of piecewise linear maps (Monfared & Durstewitz, 2020a; Sushko & Gardini, 2010), and it can be directly translated into dynamically equivalent systems of ODEs which brings further advantages for visualization and analysis (Monfared & Durstewitz, 2020b). This enables a detailed analysis of the learned model’s behavior from a DS perspective, which is particularly important in scientific contexts where we seek to understand dynamical mechanisms beyond mere prediction of future states.¹

For inferring the latent process equations simultaneously from different data sources, the PLRNN is then connected to different observation (decoder) models that embody the specific distributional properties of the respective data domains.² While this approach can be easily developed for almost any type of data model, especially in the flexible VI framework, in our examples we will focus on one of the most common multi-modal settings encountered in practice, namely when we have observed real-valued Gaussian time series $\mathbf{X} = \{\mathbf{x}_t\}$, $\mathbf{x}_t \in \mathbb{R}^{N \times 1}$, $t = 1 \dots T$, along with multi-categorical data $\mathbf{C} = \{\mathbf{c}_t\}$, where $\mathbf{c}_t \in \{0, 1\}^{K \times 1}$, $\sum_{i=1}^K c_{it} = 1$, are binary indicator vectors.

In this case, the latent model is connected to these two types of data domains through a Gaussian and a multi-categorical observation model, respectively, assuming that Gaussian (\mathbf{x}_t) and categorical (\mathbf{c}_t) observations are conditionally independent given latent state \mathbf{z}_t :

$$\mathbf{x}_t | \mathbf{z}_t \sim \mathcal{N}(\mathbf{B}\phi(\mathbf{z}_t), \boldsymbol{\Gamma}), \quad (2)$$

$$\mathbf{c}_t | \mathbf{z}_t \sim \text{Cat}(K, \boldsymbol{\pi}). \quad (3)$$

The elements of the probability vector $\boldsymbol{\pi} = (\pi_1, \dots, \pi_K)^T$

¹Schmidt et al. (2021) also discuss how to capture long-term dependencies in PLRNNs through regularization (confining the model’s eigenspectrum), and how this leads different latent states to assume different functional roles in the dynamics (although this option was not used in the present work as the time series data considered here did not have very large differences in time scales).

²On the side we note that this framework, in principle, also allows for employing more robust (non-Gaussian) distributional models for model training (e.g. Maier et al. 2017).

are generated from the latent states \mathbf{z}_t via the GLM’s natural link function with regression weights $\{\beta_i\}$, $i = 1 \dots K - 1$ (see Eq. (37), Appx. C). In Appx. C we also illustrate how to incorporate other exponential family models or mixtures thereof into our framework, but this Gaussian + categorical setting will suffice to convey the basic principles and features of our algorithm. We call the resulting model, Eq. (1-3), the *multi-modal* PLRNN (mmPLRNN) and the model defined by Eq. (1-2) the *uni-modal* PLRNN (uni-PLRNN).

We would like to infer the mmPLRNN parameters $\boldsymbol{\theta} = \{\mu_0, \mathbf{A}, \mathbf{W}, \mathbf{F}, \mathbf{h}, \mathbf{B}, \{\beta_i\}, \boldsymbol{\Gamma}, \Sigma\}$ and latent states $\mathbf{Z} = \{\mathbf{z}_t\}$ from the set of observations $\{\mathbf{X}, \mathbf{C}\}$ by maximizing the likelihood

$$\begin{aligned} p_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{C}) &= \int_{\mathbf{Z}} p_{\boldsymbol{\theta}}(\mathbf{z}_1) \prod_{t=2}^T p_{\boldsymbol{\theta}}(\mathbf{z}_t | \mathbf{z}_{t-1}) \\ &\quad \prod_{t=1}^T p_{\boldsymbol{\theta}}(\mathbf{x}_t | \mathbf{z}_t) \prod_{t=1}^T p_{\boldsymbol{\theta}}(\mathbf{c}_t | \mathbf{z}_t) d\mathbf{Z}, \end{aligned} \quad (4)$$

where we have used the conditional independence of Gaussian and categorical observations. Observations missing in one or both channels at any time point t may simply be dropped from the likelihood Eq. (4) while training. Since the log-likelihood is intractable for our model, we replace it by the evidence lower bound (ELBO, Kingma & Welling 2013; Blei et al. 2017), which in our case is

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}, q) &:= E_q [\log p_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{C}, \mathbf{Z})] + H[q(\mathbf{Z} | \mathbf{X}, \mathbf{C})] \\ &= \log p_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{C}) - \text{KL}[q(\mathbf{Z} | \mathbf{X}, \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{Z} | \mathbf{X}, \mathbf{C})] \\ &\leq \log p_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{C}), \end{aligned} \quad (5)$$

where $q(\mathbf{Z} | \mathbf{X}, \mathbf{C})$ is a proposal or variational density.

In the next two sections we will introduce both an efficient EM as well as a VI algorithm for maximizing the ELBO.³

3.2. Expectation Maximization (EM) for Model Training

It has been shown previously (Durstewitz, 2017b; Koppe et al., 2019) that the piecewise-linear structure of model Eq. (1) can be efficiently exploited in EM by a fixed-point iteration algorithm and partly analytical derivation of expectations, on which we will build here.

State Estimation In the E-step we assume, similar to a Laplace-Gaussian approximation, that the expectation value

³Note that this is a very different approach to dynamics than reservoir computing (RC; e.g. Pathak et al. 2018): While RC essentially learns to map a rich but rather fixed dynamical repertoire to the observations, here we aim to learn the underlying DS itself within the (tractable) PLRNN equations. Hence, fixing the latent and just training the observation model cannot in general work for the PLRNN (as we also explicitly checked just as a control).

$E[\mathbf{Z} | \mathbf{X}, \mathbf{C}]$ is reasonably well approximated by the mode, and solve the following maximization problem:

$$\begin{aligned} E[\mathbf{Z} | \mathbf{X}, \mathbf{C}] &\approx \mathbf{Z}^{\max} := \arg \max_{\mathbf{Z}} [\log p_{\theta}(\mathbf{Z} | \mathbf{X}, \mathbf{C})] \\ &= \arg \max_{\mathbf{Z}} [\log p_{\theta}(\mathbf{C} | \mathbf{Z}) + \log p_{\theta}(\mathbf{X} | \mathbf{Z}) + \log p_{\theta}(\mathbf{Z}) \\ &\quad - \log p_{\theta}(\mathbf{X}, \mathbf{C})] \\ &= \arg \max_{\mathbf{Z}} [\log p_{\theta}(\mathbf{C} | \mathbf{Z}) + \log p_{\theta}(\mathbf{X} | \mathbf{Z}) + \log p_{\theta}(\mathbf{Z})] . \end{aligned} \quad (6)$$

The covariance matrix of $p_{\theta}(\mathbf{Z} | \mathbf{X}, \mathbf{C})$ is then approximated by the negative inverse Hessian $(-\mathbf{H}^{\max})^{-1}$ around this maximizer, based on which all state expectations $E[\mathbf{z}]$, $E[\mathbf{zz}^T]$, $E[\phi(\mathbf{z})]$, $E[\mathbf{z}\phi(\mathbf{z})^T]$ and $E[\phi(\mathbf{z})\phi(\mathbf{z})^T]$ needed for the M-step can be computed (semi-)analytically for the PLRNN (Durstewitz, 2017b; Koppe et al., 2019).

In the original formulation of the PLRNN algorithm (Durstewitz, 2017b; Koppe et al., 2019), criterion Eq. (6) was piecewise quadratic (owing to the piecewise linear ReLU activation) and could be addressed by an efficient fixed-point-iteration algorithm. Due to the non-Gaussian terms in $p(\mathbf{C} | \mathbf{Z})$, this is no longer the case. But for any exponential family function in the decoder, Eq. (6) will remain piecewise concave (within each orthant) and can be addressed by an efficient Newton-Raphson (NR) scheme (see Appx. A.1 for details).

Parameter Estimation For parameter estimation (M-step) we assume we have all relevant moments of $q(\mathbf{Z} | \mathbf{X}, \mathbf{C})$ from the E-step and, based on this, solve the maximization problem $\theta^* := \arg \max_{\theta} \mathcal{L}(\theta, q^*)$. In the original PLRNN state space model defined by Eq. (1 - 2) one can solve this analytically and quickly in one step as a linear regression problem given all expectations in \mathbf{Z} . This is still true here for all parameters that define the latent state prior model $p_{\theta_{\text{lat}}}(\mathbf{Z})$ (Eq. (1)), $\theta_{\text{lat}} = \{\mu_0, \mathbf{A}, \mathbf{W}, \mathbf{F}, \mathbf{h}, \Sigma\}$, and those occurring within the Gaussian observation model $p_{\theta_X}(\mathbf{X} | \mathbf{Z})$, $\theta_X = \{\mathbf{B}, \Sigma\}$. However, the terms in $E[\log p(\mathbf{C} | \mathbf{Z})]$ are a bit more tricky. To separate model parameters θ from expectations in states \mathbf{z}_t , we therefore introduce another lower bound into the log-likelihood using Jensen's inequality (*) that makes the problem tractable:

$$\begin{aligned} E[\log p_{\theta}(\mathbf{C} | \mathbf{Z})] &= \\ &\sum_{t=1}^T E \left[\sum_{i=1}^{K-1} c_{it} \beta_i^T \mathbf{z}_t \right] - \sum_{t=1}^T E \left[\log \left(1 + \sum_{j=1}^{K-1} \exp(\beta_j^T \mathbf{z}_t) \right) \right] \\ &\stackrel{(*)}{\geq} \sum_{t=1}^T \sum_{i=1}^{K-1} c_{it} \beta_i^T E[\mathbf{z}_t] - \sum_{t=1}^T \log \left(1 + \sum_{j=1}^{K-1} E[\exp(\beta_j^T \mathbf{z}_t)] \right) . \end{aligned} \quad (7)$$

Further noting that states \mathbf{z}_t are conditionally Gaussian, we can use the moment-generating function of the Gaussian

(see also Smith & Brown 2003) to reshape Eq. (7) as

$$\begin{aligned} f(\beta) &:= \sum_{t=1}^T \sum_{i=1}^{K-1} c_{it} \beta_i^T E[\mathbf{z}_t] \\ &- \sum_{t=1}^T \log \left[1 + \sum_{j=1}^{K-1} \exp \left(\beta_j^T E[\mathbf{z}_t] + \frac{\beta_j^T \text{Cov}(\mathbf{z}_t) \beta_j}{2} \right) \right] . \end{aligned} \quad (8)$$

This is a concave function again in parameters β that only requires expectations $E[\mathbf{z}_t]$ and $E[\mathbf{z}_t \mathbf{z}_t^T]$ from the E-step, which can hence be solved quickly and efficiently by NR iterations (see Appx. A.1).

Since all exponential family distributions, as well as sums of exponential family models, have concave log-likelihoods (Kandala et al., 2001), one can always employ the NR scheme for the E- and M-steps as in Eq. (17) and (18), as long as the distributional parameters are connected to the latent states through the natural link function. This makes the above algorithm more generally applicable (beyond just Gaussian and categorical observations). For more details on training see Appx. A.1 & Appx. A.2, *Stepwise training protocol*.

3.3. Variational Inference for Model Training

The EM algorithm for PLRNN inference has been shown to efficiently work with small data sets (Koppe et al., 2019), but it lacks flexibility (other than exponential family distributions may be harder to accommodate). An alternative way to optimize expression (5) is VI, whereby $q_{\zeta}(\mathbf{Z} | \mathbf{X}, \mathbf{C})$ becomes a parameterized family of variational densities for approximating the true posterior. We model the approximate posterior by a multivariate Gaussian,

$$q_{\zeta}(\mathbf{Z} | \mathbf{X}, \mathbf{C}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{Z}}(\mathbf{X}, \mathbf{C}), \boldsymbol{\Lambda}_{\mathbf{Z}}(\mathbf{X}, \mathbf{C})) , \quad (9)$$

where the mean $\boldsymbol{\mu}_{\mathbf{Z}}(\mathbf{X}, \mathbf{C}) \in \mathbb{R}^{MT \times 1}$ and covariance matrix $\boldsymbol{\Lambda}_{\mathbf{Z}}(\mathbf{X}, \mathbf{C}) \in \mathbb{R}^{MT \times MT}$ are parameterized by neural networks with parameters $\zeta = \{\zeta_{\mu}, \zeta_{\Lambda}\}$. Specifically, instead of parameterizing a full covariance Λ directly, we follow (Archer et al., 2015) and parameterize the $M \times M$ on- and off-diagonal blocks of the Hessian $\mathbf{H} = -\boldsymbol{\Lambda}_{\mathbf{Z}}^{-1}$ by neural networks, exploiting its block tri-diagonal structure owing to the Markovian latent model assumptions (see Appx. A.3 for more details).

Joint optimization of variational (ζ) and generative (θ) model parameters is performed via Stochastic Gradient Variational Bayes (SGVB) using the re-parameterization trick for the model's random variables (Kingma & Welling, 2013; Rezende et al., 2014). We chose Adam (Kingma & Ba, 2015) with learning rate $\alpha = 10^{-3}$.

All code produced here is available at github.com/DurstewitzLab/mmPLRNN.

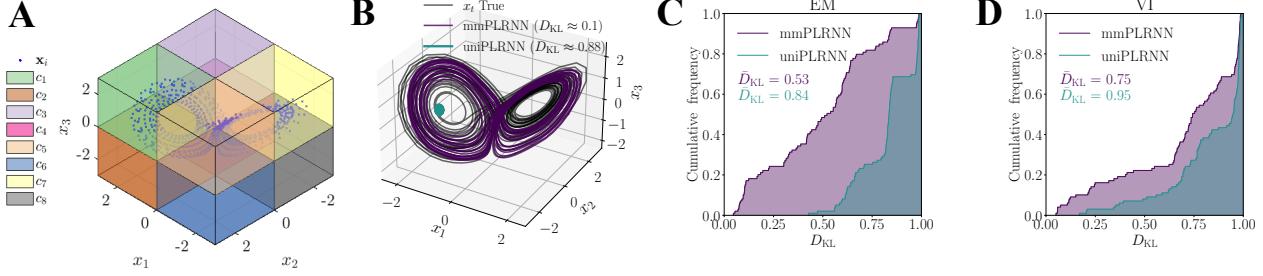


Figure 2: Improving DS reconstruction with multi-modal data when continuous observations are too noisy. A) Experimental setup with Gaussian and categorical information. B) Example of successful DS reconstruction with multi-modal (purple) but not uni-modal (cyan) PLRNN. Black trajectory = ground truth. C) Cumulative performance histograms ($n = 100$ runs) in terms of normalized Kullback-Leibler divergence D_{KL}/D_{KL}^{\max} between true and generated attractor geometries for uni- vs. multi-modal PLRNN produced by the EM algorithm. \bar{D}_{KL} indicates the median. D) Same for models trained through VI.

4. Results

We first evaluate the algorithm’s ability to combine information from different, distinct data streams into a common latent nonlinear DS model on two purpose-designed ground truth systems. For these we produce both continuous Gaussian and categorical information from the Lorenz ODE system within its chaotic regime (Lorenz, 1963), a popular benchmark system for testing DS reconstruction. We then probe our algorithm on experimental data consisting of simultaneous functional Magnetic Resonance Imaging (fMRI) recordings of different brain regions and (categorical) behavioral data from healthy subjects performing a working memory task (Koppe et al., 2014). Detailed information on hyper-parameter settings for all methods and experiments is collected in Appx. A.4.

4.1. Benchmarks: Noisy or Incomplete Lorenz System with Gaussian and Categorical Observations

The 3D-Lorenz system is defined by the set of Eqs. (32) (see Appx. B.1) where we have added a Gaussian dynamical (process) noise term $d\epsilon(t) \sim \mathcal{N}(0, 0.0025 \times dt\mathbf{I})$ when integrating the equations, making this a system of stochastic differential equations. State trajectories $\mathbf{x}_t = (x_1, x_2, x_3)^T$ were generated from this system (Fig. 2A) using fourth-order Runge-Kutta numerical integration (Press et al., 2007). Generated trajectories were further standardized (centered and scaled to unit variance on each dimension) and blurred by adding a relatively large amount of Gaussian observation noise $\eta_t \sim \mathcal{N}(0, 0.1 \times \mathbf{I})$, strongly degrading the information about the underlying DS that could be obtained from the continuous Gaussian observations alone. This emulates a natural scenario where one information channel on its own may be too noisy to enable identification of the underlying system. In addition to these Gaussian observations, we provide categorical information about the system’s dynamics by indicating in which of the eight orthants the system’s current

state is in (Fig. 2A), i.e., in the form of an 8-dimensional indicator vector $\mathbf{c}_t = (c_{1t}, \dots, c_{8t})^T$, where $c_{it} = 1$ for the $(I[x_{1t} > 0]2^0 + I[x_{2t} > 0]2^1 + I[x_{3t} > 0]2^2 + 1)^{th}$ bit and 0 otherwise. The mmPLRNN algorithm had access to only one relatively short time series $\{\mathbf{x}_t, \mathbf{c}_t\}$, $t = 1\dots T$, of length $T = 1000$ to infer the underlying DS, using $M = 15$ latent states (see Appx. A.4).

To evaluate the quality of DS reconstruction, new trajectories were sampled from the once trained generative model $p_{\theta}(\mathbf{X}, \mathbf{C}, \mathbf{Z})$, i.e. new latent state trajectories were first drawn from the model prior $\mathbf{Z} \sim p_{\theta_{lat}}(\mathbf{Z})$ defined by the latent process Eq. (1), from which time series observations $\mathbf{X} \sim p_{\theta_X}(\mathbf{X} | \mathbf{Z})$ and $\mathbf{C} \sim p_{\theta_C}(\mathbf{C} | \mathbf{Z})$ were produced according to the learned observation models. Fig. 2B provides an example of successful reconstruction of the Lorenz system, i.e. where the mmPLRNN’s intrinsic dynamics captures well the butterfly-wing structure of the chaotic Lorenz attractor.

To quantify reconstruction quality, we used a previously introduced Kullback-Leibler measure for the agreement between true, $\hat{p}_{true}(\mathbf{x})$, and model-generated, $\hat{p}_{gen}(\mathbf{x} | \mathbf{z})$, *attractor geometries* (Koppe et al., 2019) (see Eq. (33) in Appx. B.2). Importantly, this measure assesses the agreement across space, not time: As pointed out in (Woods, 2010; Koppe et al., 2019), trajectories from chaotic systems not started from exactly the same initial condition exponentially diverge, potentially leading to ultimately highly dissimilar time series with large MSE deviation, even though they may have been generated by the very same DS (see Fig. 2 in Koppe et al. 2019). In contrast, (time-) invariant properties like attractor geometries should be preserved. As shown in Fig. 2C-D, attractor reconstructions as assessed by this measure strongly improve when the algorithm has access to the categorical information on top of the Gaussian time series, in contrast to when only the latter were available. This was true regardless of whether the mmPLRNN was

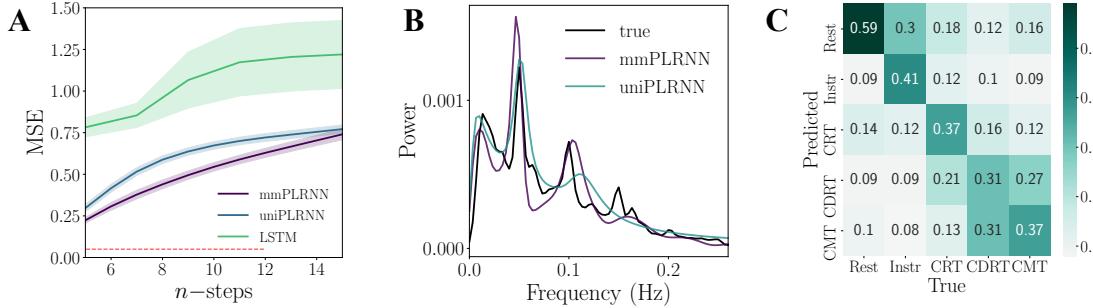


Figure 3: mmPLRNN trained on simultaneous BOLD recordings and categorical data from fMRI experiments. A) MSE for n -step ahead predictions for uni- vs. mmPLRNN, and for LSTMs trained with Adam and learning rate $\alpha = 0.005$. fMRI sampling rate was $1/3$ Hz, so one step corresponds to 3 s. Error shadings indicate SEM. For the indicated time steps (red dots), the difference between uni- and mmPLRNN is significant ($p < .05$) in Tukey post-hoc tests. Note that for a chaotic system, multimodal gains in predictability are expected to vanish for larger time steps due to the exponential divergence of trajectories. B) Example reproduction of power spectra. C) Confusion matrix of predicted vs. true class labels on test sets. Base rates of classes were 0.32 (Rest), 0.125 (Instr), 0.185 (CRT), 0.185 (CDRT), 0.185 (CMT).

trained by EM or VI, although for EM the improvement was more pronounced and reconstructions were better on average (this advantage of EM disappeared, however, if VI was allowed access to longer training sequences, $T = 2500$: $\bar{D}_{KL} \approx 0.49$ ($n = 40$), see also Fig. 6). This demonstrates that the mmPLRNN can strongly enhance DS identification by supplementing the highly noisy trajectory information by categorical data, even though, and importantly, these do not provide *quantitative* information about the dynamics (in particular, no information about attractor geometry or topology). Of course, as noise levels become very low, information from a continuous-Gaussian channel alone may enable sufficiently good reconstructions (as further explored in Fig. 6).

As a second test case, we studied whether additional categorical information could also help to identify the chaotic Lorenz system when one of its dynamical variables (x_2) was missing from the observations, i.e., only $\mathbf{x}_t^{\text{red}} = (x_{1t}, x_{3t})^T$ was provided for training. This is indeed the case, as reported in Appx. B.6, Fig. 5.

4.2. Empirical example: DS inference from Neurophysiological and Task Label Data

For probing the mmPLRNN on real data, we chose a data set consisting of fMRI recordings (which assess the so-called Blood-Oxygenation-Level-Dependent, BOLD, signal) taken from human subjects while they performed simple working memory and control tasks. The details of the experimental setup are not overly important here and are given in Koppe et al. (2014) and briefly summarized in Appx. B.3. $N = 20$ brain regions (from $l = 21$ subjects, see Appx. B.3) were selected for analysis, yielding continuous time series data $\mathbf{X} = \{\mathbf{x}_t\}$, $\mathbf{x}_t = (x_{1,t}, \dots, x_{20,t})^T$ for each

subject. Of note, BOLD time series were relatively short ($T = 360$) and hence extracting reasonable dynamics from *single subjects* is quite challenging. In fact, for this type of very sparse data only the more efficient EM algorithm tended to produce successful reconstructions, and hence only these are reported here. As categorical data we defined the five major task stages subjects underwent during the experiment ('Rest', 'Instruction', 'Choice Reaction Task', 'Continuous Delayed Response Task', 'Continuous Matching Task'), and endowed each time step with a categorical label $\mathbf{c}_t \in \{0, 1\}^5$ accordingly.⁴

As in the Lorenz benchmark setups, we first studied whether including categorical information would help the algorithm to produce better reconstructions and predictions as compared to when only BOLD time series were provided. Fig. 3A shows the ahead-prediction error for $n \in \{5 \dots 15\}$ future time steps, and for both a uni-modal PLRNN, trained only on the BOLD signals, and the mmPLRNN which consumed class labels in addition (both trained with $M = 20$ states; for comparison, also predictions produced by a LSTM with the same number of latent states are shown). There is a consistent and overall highly significant (repeated ANOVA, $p < .001$) increase in performance (up to 26%) for the mmPLRNN across all prediction time steps. The mmPLRNN also produced on average significantly better geometrical reconstructions as revealed by our Kullback-Leibler criterion (unnormalized $\langle D_{mm} \rangle \approx 1.09 \pm 0.03 < \langle D_{uni} \rangle \approx 2.56 \pm 0.03$, $p < 10^{-4}$, Wilcoxon ranksum test). Thus, the additional categorical information indeed significantly helped to reconstruct the underlying system. The

⁴We stress that these different task stages did *not* differ in the type of presented stimuli or required responses, i.e. in their 'external inputs', but rather tapped into different cognitive processes.

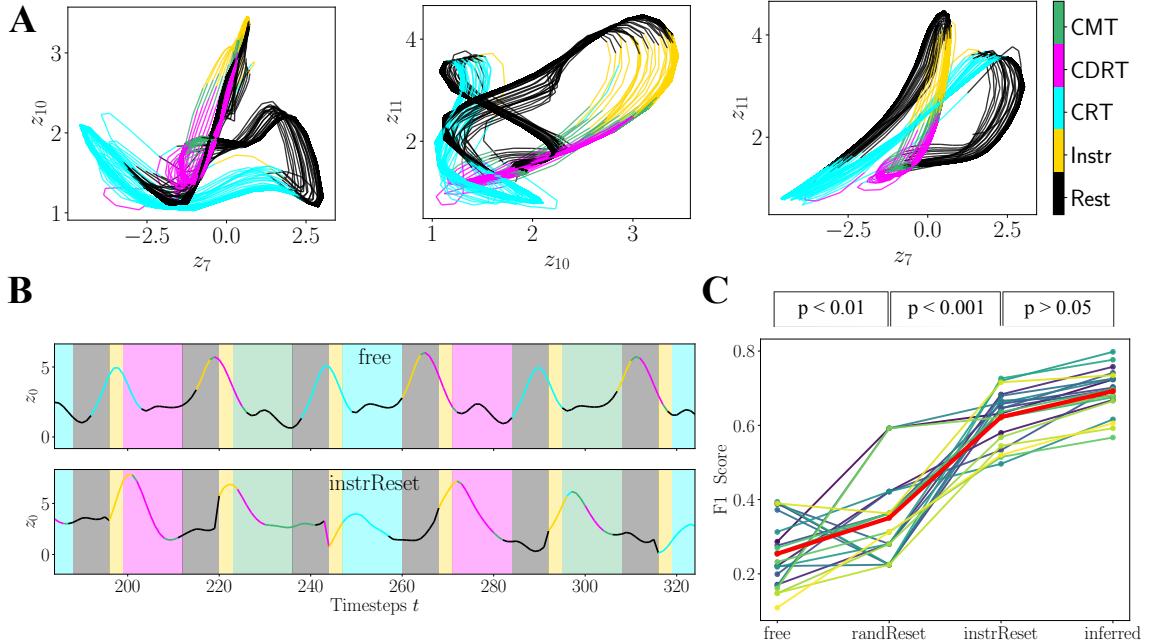


Figure 4: Neural dynamics underlying task stages. A) Association between predicted class labels (color coding) and learned BOLD dynamics. Shown are 2d subspaces of an mmPLRNN’s generated state space. Subspaces chosen for display were selected according to Fisher’s discriminant criterion. B) Top: Freely generated latent activity (initialized only once with the inferred state at the beginning of the experiment), color-coded according to task stage as in A. True task stages indicated by the pale background colors. Bottom: Same, but with generated latents reset to inferred values upon each new instruction phase. C) Summary statistics across $l = 21$ subjects, comparing task stage decoding from simulated latent activity initialized only at experiment onset (“free”), simulated activity reset at 15 random time bins (“randReset”), simulated activity reset at the 15 instruction onsets (“instrReset”), and fully inferred latent states (“inferred”, i.e. the conditional means $E[\mathbf{Z} | \mathbf{X}]$). Red line = group means. Repeated measures ANOVA ($F \approx 138.56, p < 10^{-5}$) with Tukey posthoc-tests as indicated.

example of true and reconstructed power spectra in Fig. 3B furthermore confirms that the mmPLRNN has learned to generate (i.e., freely simulate) time series which exhibit the same major temporal signatures (peaks in the spectrum) as the real data. Hence, also for this empirical example the mmPLRNN seems to exploit both data modalities to achieve the best reconstruction. This is an important and non-trivial result, as it confirms that even in this empirical scenario purely categorical information may help in guiding the algorithm toward better approximations of the underlying neural dynamics.

Furthermore, we tested cross-modal inference, i.e. whether the trained mmPLRNN would be able to predict class labels from BOLD signals alone on left-out test data. While here this mainly serves to examine whether cross-modal links have been built within the model’s latent space, it is also a relevant application case. Specifically, we ran a cross-validation protocol where each 20% segment of the time series was left out in turn for training, and unseen class labels were predicted on these left-out test sets (see Appx. B.3 for details). Fig. 3C summarizes the agreement

between true and predicted behavioral class labels across all test sets from successful training runs (see Appx. B.3) in a confusion matrix. These results were on par with those produced by various classifiers trained *directly* on the same 20-dimensional BOLD signal vectors and associated class labels as used for the mmPLRNN (multi-class F1 scores for logistic regression: ≈ 0.47 , linear discriminant analysis: ≈ 0.48 , support vector machines (RBF kernel, $C = 1$): ≈ 0.47 , mmPLRNN: ≈ 0.45). This confirms that the mmPLRNN has learned the connections between the two data modalities within its latent space in an about optimal manner, i.e., without much loss of information as judged by this comparison.

Of course, in practice we would not use the mmPLRNN merely for cross-modal prediction. Rather, the main purpose of this approach is that we can now harvest the trained model and common latent representation to investigate the *dynamical mechanisms* of the observed BOLD signals and cross-modal links. In general, properly trained mmPLRNNs exhibited complex cycles (often chaotic oscillators, Fig. 4A) as their limiting behavior (i.e., attractors) that mimic

the temporal structure of the BOLD signal. For the example in Fig. 4A we analytically computed the PLRNN’s maximum Lyapunov exponent as $\lambda_{\max} \approx 0.009$ (attesting its chaotic nature, see Appx. B.5). As the example shows, different task stages seem to be associated with different subcycles or phases of the chaotic oscillator. Across all subjects, *freely running* mmPLRNN oscillators⁵ initially (at the start of an experiment) predicted task stages quite well, but then started to run out of phase with the experimental stages (significantly better agreement with true class labels in 1st ($F_1 \approx 0.38$) compared to 2nd ($F_1 \approx 0.17$) and 3rd ($F_1 \approx 0.16$) thirds of time series; $F \approx 30.68$, $p < 10^{-5}$). This is expected since, unlike the real experimental subjects who were updated with each new instruction phase, the freely simulated mmPLRNN does not receive any task-related information but simply follows its internal dynamics. Indeed, resetting the intrinsic PLRNN oscillators at the beginning of each instruction phase to the inferred latent state (i.e., posterior estimate $E[\mathbf{z}_t | \mathbf{X}]$) significantly improved the alignment with the experimental task stages (Fig. 4B; Fig. 4C, “instrReset”); in particular, significantly more so than just resetting the PLRNN oscillators to inferred values at arbitrary (but similarly spaced) time points throughout the experiment (Fig. 4C, “randReset”). In contrast, replacing *all* simulated latent states by inferred values did not yield a significant improvement in task alignment anymore (Fig. 4C, “inferred”). Hence, the mmPLRNN explains the links between BOLD activity and task stages through a complex oscillator that is differently initialized upon each distinct instruction phase.

While multivariate classifiers have long been used for reading out sensory or cognitive representations from fMRI activity (Haynes & Rees, 2006; Haynes, 2015), methods like the mmPLRNN therefore enable to reveal much more specifically, in terms of DS mechanisms, how BOLD dynamics and mental processes are linked. Neural oscillations, in particular, play a huge role in cognition and memory formation (Buzsáki, 2006). The functional significance of slower oscillations as detectable with fMRI is as yet unclear (Drew et al., 2020; Lewis et al., 2016), however, where the present methods may help to improve our understanding. Exploring these possibilities further for other fMRI task conditions or subject groups (patients) would be an interesting future direction.

5. Conclusions

In this work we introduced a new algorithm for nonlinear DS reconstruction, the mmPLRNN, that draws on several data channels described by different distributional models. By DS reconstruction here we meant that the trained system

approximates the true underlying DS in a generative sense, i.e. such that after training trajectories drawn (simulated) from the latent process would produce the same state space behavior and invariant properties (like attractor geometries) as the observed system. Although various approaches toward this goal have been introduced before (Brunton et al., 2016; Champion et al., 2019; Duncker et al., 2019; Lu et al., 2017a; Durstewitz, 2017b; Koppe et al., 2019; Razaghi & Paninski, 2019), to our knowledge the mmPLRNN is the first such system that can integrate different types of data modalities for this purpose. We developed both an EM- and a VI-based variant of the basic algorithm, and demonstrated that the mmPLRNN will use categorical (or other, see Appx. C) information to fill in for information too noisy or missing in the Gaussian channel, i.e. will effectively combine the different information sources to infer the underlying DS. We also showcased the mmPLRNN on a neuroscientific dataset consisting of simultaneous fMRI recordings and behavioral task labels, and showed how it could be used to gain insight into the neural dynamics underlying BOLD-class label associations.

Both the EM- and VI-based algorithms have their own advantages and drawbacks: VI scales better with problem size than EM, as it can be efficiently trained through gradient descent using the reparameterization trick (Kingma & Welling, 2013; Rezende et al., 2014; Krishnan et al., 2015; Archer et al., 2015). It is also more flexible as it can more easily accommodate a larger variety of distributional models. For the EM-based mmPLRNN, on the other hand, although the steps outlined here for categorical data are fairly easy to extend to other exponential family models, more complex distributional models would require additional thought and possibly hand-crafted solutions. On the upside, the EM algorithm can more efficiently deal with smaller scale problems as often encountered in physical or biological experiments (like the fMRI data studied here), and provides higher accuracy estimates that may be preferable in scientific or medical settings. This is because for the PLRNN our EM can – given $p(\mathbf{Z} | \mathbf{X}, \mathbf{C})$ – compute many of the parameters and expectations analytically and, in contrast to VI, is second-order even in its numerical parts (see Appx. A.1). An interesting question for future investigation is how much information about an underlying DS can be recovered *solely* from non-continuous and non-Gaussian data, like categorical or count series.

6. Acknowledgements

This work was funded by grants from the German Research Foundation (DFG) within the CRC TRR-265 (A06 & B08) to DD and GK, and through Du 354/10-1 (DFG) to DD.

⁵By this we mean $E[\mathbf{Z}]$ computed by forward-iterating Eq. 1 in time from μ_0 as inferred from the very first time step.

References

- Abarbanel, H. *Predicting the Future: Completing Models of Observed Complex Systems (Understanding Complex Systems)*. Springer, 2013. ISBN 978-1-4614-7218-6.
- 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 Stoyanov, D., Taylor, Z., Kia, S. M., Oguz, I., Reyes, M., Martel, A. L., Maier-Hein, L., Marquand, A. F., Duchesnay, E., Löfstedt, T., Landman, B. A., Cardoso, M. J., Silva, C. A., Pereira, S., and Meier, R. (eds.), *Understanding and Interpreting Machine Learning in Medical Image Computing Applications - First International Workshops MLCN 2018, DLF 2018, and iMIC 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16-20, 2018, Proceedings*, volume 11038 of *Lecture Notes in Computer Science*, pp. 15–23. Springer, 2018. doi: 10.1007/978-3-030-02628-8_2. URL https://doi.org/10.1007/978-3-030-02628-8_2.
- Archer, E., Park, I. M., Buesing, L., Cunningham, J., and Paninski, L. Black box variational inference for state space models. *ArXiv*, 2015.
- Ayed, I., de Bézenac, E., Pajot, A., Brajard, J., and Gallinari, P. Learning dynamical systems from partial observations. *arXiv*, 2019.
- Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M. Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 2018.
- Bhagwat, N., Viviano, J. D., Voineskos, A. N., and and, M. M. C. Modeling and prediction of clinical symptom trajectories in alzheimer's disease using longitudinal data. *PLOS Computational Biology*, 14(9):e1006376, sep 2018. doi: 10.1371/journal.pcbi.1006376.
- Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, April 2017. doi: 10.1080/01621459.2017.1285773. URL <https://doi.org/10.1080/01621459.2017.1285773>.
- Brunton, S. L. and Kutz, J. N. *Data-driven science and engineering*. Cambridge University Press, Cambridge, England, February 2019.
- 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*, 113(15):3932–3937, mar 2016. doi: 10.1073/pnas.1517384113.
- Buzsáki, G. *Rhythms of the Brain*. Oxford University Press, October 2006. doi: 10.1093/acprof:oso/9780195301069.001.0001. URL <https://doi.org/10.1093/acprof:oso/9780195301069.001.0001>.
- Champion, K., Lusch, B., Kutz, J. N., and Brunton, S. L. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116(45):22445–22451, October 2019. doi: 10.1073/pnas.1906995116. URL <https://doi.org/10.1073/pnas.1906995116>.
- Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL <https://proceedings.neurips.cc/paper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf>.
- Chen, R. T. Q., Amos, B., and Nickel, M. Learning neural event functions for ordinary differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=kW_zpEmMLdP.
- Chen, S., Shojaie, A., and Witten, D. M. Network reconstruction from high-dimensional ordinary differential equations. *Journal of the American Statistical Association*, 112(520):1697–1707, aug 2017. doi: 10.1080/01621459.2016.1229197.
- Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2(4):303–314, dec 1989. doi: 10.1007/bf02551274.
- Dezfouli, A., Morris, R., Ramos, F., Dayan, P., and Balleine, B. W. Integrated accounts of behavioral and neuroimaging data using flexible recurrent neural network models. *NIPS*, may 2018. doi: 10.1101/328849.
- Drew, P. J., Mateo, C., Turner, K. L., Yu, X., and Kleinfield, D. Ultra-slow oscillations in fMRI and resting-state connectivity: Neuronal and vascular contributions and technical confounds. *Neuron*, 107(5):782–804, 2020. ISSN 0896-6273. doi: <https://doi.org/10.1016/j.neuron.2020.07.020>. URL <https://www.sciencedirect.com/science/article/pii/S089662732030564X>.
- Duncker, L., Bohner, G., Boussard, J., and Sahani, M. Learning interpretable continuous-time models of latent stochastic dynamical systems. volume 97 of *Proceedings of Machine Learning Research*, pp. 1726–1734. PMLR, 09–15 Jun 2019. URL <http://proceedings.mlr.press/v97/duncker19a.html>.
- Durstewitz, D. Self-organizing neural integrator predicts interval times through climbing activity. *The Journal of Neuroscience*, 23(12):5342–5353, June 2003. doi: 10.1523/jneurosci.23-12-05342.2003. URL <https://doi.org/10.1523/jneurosci.23-12-05342.2003>.
- Durstewitz, D. *Advanced data analysis in Neuroscience*. Springer, 2017a. ISBN 978-3-319-59974-8.
- Durstewitz, D. A state space approach for piecewise-linear recurrent neural networks for identifying computational dynamics from neural measurements. *PLOS*, 13(6):e1005542, jun 2017b. doi: 10.1371/journal.pcbi.1005542.
- Durstewitz, D., Seamans, J. K., and Sejnowski, T. J. Neurocomputational models of working memory. *Nature Neuroscience*, 3 (S11):1184–1191, nov 2000. doi: 10.1038/81460.
- Funahashi, K. and Nakamura, Y. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, 6(6):801–806, jan 1993. doi: 10.1016/s0893-6080(05)80125-x.

- Gorbach, N. S., Bauer, S., and Buhmann, J. M. Scalable variational inference for dynamical systems. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL <https://proceedings.neurips.cc/paper/2017/file/e71e5cd119bbc5797164fb0cd7fd94a4-Paper.pdf>.
- Halpern, D., Tubridy, S., Wang, H. Y., Gasser, C., Popp, P. J. O., Davachi, L., and Gureckis, T. M. Knowledge tracing using the brain. *PsyArXiv*, apr 2018. doi: 10.31234/osf.io/fmj48.
- Hanson, J. and Raginsky, M. Universal simulation of stable dynamical systems by recurrent neural nets. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120 of *Proceedings of Machine Learning Research*, pp. 384–392, The Cloud, 10–11 Jun 2020. PMLR. URL <http://proceedings.mlr.press/v120/hanson20a.html>.
- Haynes, J.-D. A primer on pattern-based approaches to fmri: Principles, pitfalls, and perspectives. *Neuron*, 87(2):257–270, 2015. ISSN 0896-6273. doi: <https://doi.org/10.1016/j.neuron.2015.05.025>. URL <https://www.sciencedirect.com/science/article/pii/S0896627315004328>.
- Haynes, J.-D. and Rees, G. Decoding mental states from brain activity in humans. *Nature Reviews Neuroscience*, 7(7):523–534, July 2006. doi: 10.1038/nrn1931. URL <https://doi.org/10.1038/nrn1931>.
- Hernández, C. X., Wayment-Steele, H. K., Sultan, M. M., Husic, B. E., and Pande, V. S. Variational encoding of complex dynamics. *Phys. Rev. E*, 97:062412, Jun 2018. doi: 10.1103/PhysRevE.97.062412. URL <https://link.aps.org/doi/10.1103/PhysRevE.97.062412>.
- Hernandez, D., Moretti, A. K., Wei, Z., Saxena, S., Cunningham, J., and Paninski, L. Nonlinear evolution via spatially-dependent linear dynamics for electrophysiology and calcium data. *Neurons, Behavior, Data analysis, and Theory (NBDT)*, 2018. URL [arXiv:1811.02459](https://arxiv.org/abs/1811.02459).
- Hertag, L., Durstewitz, D., and Brunel, N. Analytical approximations of the firing rate of an adaptive exponential integrate-and-fire neuron in the presence of synaptic noise. *Frontiers in Computational Neuroscience*, 8, sep 2014. doi: 10.3389/fncom.2014.00116.
- Hochreiter, S., Heusel, M., and Obermayer, K. Fast model-based protein homology detection without alignment. *Bioinformatics*, 23(14):1728–1736, 05 2007. ISSN 1367-4803. doi: 10.1093/bioinformatics/btm247. URL <https://doi.org/10.1093/bioinformatics/btm247>.
- Iakovlev, V., Heinonen, M., and Lähdesmäki, H. Learning continuous-time {pde}s from sparse data with graph neural networks. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=aUX5Plaq7Oy>.
- Kalman, R. E. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(Series D):35–45, 1960.
- Kandala, N. B., Lang, S., Klasen, S., and Fahrmeir, L. Semiparametric analysis of the socio-demographic and spatial determinants of undernutrition in two african countries, 2001. URL <http://nbn-resolving.de/urn/resolver.pl?urn=nbn:de:bvb:19-epub-1626-2>.
- Kimura, M. and Nakano, R. Learning dynamical systems by recurrent neural networks from orbits. *Neural Networks*, 11(9):1589–1599, dec 1998. doi: 10.1016/s0893-6080(98)00098-7.
- Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings*, 2015. URL <http://arxiv.org/abs/1412.6980>.
- Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. *Arxiv*, 2013.
- Koch, C. and Sevèg, I. *Methods in Neuronal Modeling (Computational Neuroscience Series): From Ions to Networks (Computational Neuroscience) Second Edition*. A Bradford Book, 2003. ISBN 9780262517133.
- 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, 07 2014. 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, nov 2018. 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, aug 2019.
- Krishnan, R. G., Shalit, U., and Sontag, D. Deep kalman filters. *arXiv*, 2015.
- Kurle, R., Günemann, S., and der Smagt, P. V. Multi-source neural variational inference. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:4114–4121, July 2019. doi: 10.1609/aaai.v33i01.33014114. URL <https://doi.org/10.1609/aaai.v33i01.33014114>.
- Kusner, M. J., Paige, B., and Hernández-Lobato, J. M. Grammar variational autoencoder. volume 70 of *Proceedings of Machine Learning Research*, pp. 1945–1954. PMLR, 06–11 Aug 2017. URL <http://proceedings.mlr.press/v70/kusner17a.html>.
- Lahat, D., Adali, T., and Jutten, C. Multimodal data fusion: An overview of methods, challenges, and prospects. *Proceedings of the IEEE*, 103(9):1449–1477, sep 2015. doi: 10.1109/jproc.2015.2460697.
- Lambert, D. Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, 34:1–14, 02 1992. doi: 10.1080/00401706.1992.10485228.
- Lee, J.-T., Jain, M., Park, H., and Yun, S. Cross-attentional audio-visual fusion for weakly-supervised action localization. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=hWr3e3r-oH5>.
- Lewis, L. D., Setsompop, K., Rosen, B. R., and Polimeni, J. R. Fast fMRI can detect oscillatory neural activity in humans. *Proceedings of the National Academy of Sciences*, 113(43):E6679–E6685, October 2016. doi: 10.1073/pnas.1608117113. URL <https://doi.org/10.1073/pnas.1608117113>.

- 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, jul 2015. doi: 10.1109/tcbb.2014.2377729.
- Lin, H. and Jegelka, S. ResNet with one-neuron hidden layers is a universal approximator. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL <https://proceedings.neurips.cc/paper/2018/file/03bfc1d4783966c69cc6aef8247e0103-Paper.pdf>.
- Lorenz, E. N. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, March 1963. doi: 10.1175/1520-0469(1963)020<0130:dnf>2.0.co;2. URL [https://doi.org/10.1175/1520-0469\(1963\)020<0130:dnf>2.0.co;2](https://doi.org/10.1175/1520-0469(1963)020<0130:dnf>2.0.co;2).
- Lu, Z., Pathak, J., Hunt, B., Girvan, M., Brockett, R., and Ott, E. Reservoir observers: Model-free inference of unmeasured variables in chaotic systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 27(4):041102, apr 2017a. doi: 10.1063/1.4979665.
- Lu, Z., Pu, H., Wang, F., Hu, Z., and Wang, L. The expressive power of neural networks: A view from the width. *NIPS*, 2017b.
- Machens, C. K., Romo, R., and Brody, C. D. Flexible control of mutual inhibition: A neural model of two-interval discrimination. *Science*, 307(5712):1121–1124, February 2005. doi: 10.1126/science.1104171. URL <https://doi.org/10.1126/science.1104171>.
- Maier, C., Loos, C., and Hasenauer, J. Robust parameter estimation for dynamical systems from outlier-corrupted data. *Bioinformatics*, pp. btw703, January 2017. doi: 10.1093/bioinformatics/btw703. URL <https://doi.org/10.1093/bioinformatics/btw703>.
- Meeds, T., Roeder, G., Grant, P., Phillips, A., and Dalchau, N. Efficient amortised Bayesian inference for hierarchical and nonlinear dynamical systems. volume 97 of *Proceedings of Machine Learning Research*, pp. 4445–4455. PMLR, 09–15 Jun 2019. URL <http://proceedings.mlr.press/v97/meeds19a.html>.
- Miotto, R., Li, L., Kidd, B. A., and Dudley, J. T. Deep patient: An unsupervised representation to predict the future of patients from the electronic health records. *Scientific Reports*, 6(1), may 2016. doi: 10.1038/srep26094.
- Molano-Mazon, M., Onken, A., Piasini, E., and Panzeri, S. Synthesizing realistic neural population activity patterns using generative adversarial networks. In *International Conference on Learning Representations*, 2018. URL <https://openreview.net/forum?id=r1VVsebAZ>.
- Monfared, Z. and Durstewitz, D. Existence of n-cycles and border-collision bifurcations in piecewise-linear continuous maps with applications to recurrent neural networks. *Nonlinear Dynamics*, 101(2):1037–1052, July 2020a. doi: 10.1007/s11071-020-05841-x. URL <https://doi.org/10.1007/s11071-020-05841-x>.
- Monfared, Z. and Durstewitz, D. Transformation of ReLU-based recurrent neural networks from discrete-time to continuous-time. volume 119 of *Proceedings of Machine Learning Research*, pp. 6999–7009. PMLR, 13–18 Jul 2020b. URL <http://proceedings.mlr.press/v119/monfared20a.html>.
- Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML ’11, pp. 689–696, New York, NY, USA, June 2011. ACM. ISBN 978-1-4503-0619-5.
- Owen, A. M., McMillan, K. M., Laird, A. R., and Bullmore, E. N-back working memory paradigm: A meta-analysis of normative functional neuroimaging studies. *Human Brain Mapping*, 25(1):46–59, 2005. doi: 10.1002/hbm.20131. URL <https://doi.org/10.1002/hbm.20131>.
- Pandarinath, C., O’Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., Trautmann, E. M., Kaufman, M. T., Ryu, S. I., Hochberg, L. R., Henderson, J. M., Shenoy, K. V., Abbott, L. F., and Sussillo, D. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15(10):805–815, September 2018. doi: 10.1038/s41592-018-0109-9. URL <https://doi.org/10.1038/s41592-018-0109-9>.
- Paninski, L., Ahmadian, Y., Ferreira, D. G., Koyama, S., Rad, K. R., Vidne, M., Vogelstein, J., and Wu, W. A new look at state-space models for neural data. *Journal of Computational Neuroscience*, 29(1-2):107–126, aug 2009. doi: 10.1007/s10827-009-0179-x.
- 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. *Physical Review Letters*, 120(2), January 2018. doi: 10.1103/physrevlett.120.024102. URL <https://doi.org/10.1103/physrevlett.120.024102>.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. *Numerical Recipes*. Cambridge University Pr., 2007. ISBN 0521880688.
- Purdon, P., Lamus, C., Hamalainen, M., and Brown, E. A state space approach to multimodal integration of simultaneously recorded EEG and fMRI. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, mar 2010. doi: 10.1109/icassp.2010.5494906.
- Radu, V., Lane, N. D., Bhattacharya, S., Mascolo, C., Marina, M. K., and Kawsar, F. Towards multimodal deep learning for activity recognition on mobile devices. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct - UbiComp16*. ACM Press, 2016. doi: 10.1145/2968219.2971461.
- Raissi, M. Deep hidden physics models: Deep learning of nonlinear partial differential equations. *Journal of Machine Learning Research*, 19(25):1–24, 2018. URL <http://jmlr.org/papers/v19/18-046.html>.
- Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P. J., Liu, X., Marcus, J., Sun, M., Sundberg, P., Yee, H., Zhang, K., Zhang, Y., Flores, G., Duggan, G. E., Irvine, J., Le, Q., Litsch, K., Mossin, A., Tansuwan, J., Wang, D., Wexler, J.,

- Wilson, J., Ludwig, D., Volchenboum, S. L., Chou, K., Pearson, M., Madabushi, S., Shah, N. H., Butte, A. J., Howell, M. D., Cui, C., Corrado, G. S., and Dean, J. Scalable and accurate deep learning with electronic health records. *npj Digital Medicine*, 1(1), may 2018. doi: 10.1038/s41746-018-0029-1.
- Razaghi, H. S. and Paninski, L. Filtering normalizing flows. In *4th workshop on Bayesian Deep Learning (NeurIPS 2019)*, 2019.
- Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. volume 32 of *Proceedings of Machine Learning Research*, pp. 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR. URL <http://proceedings.mlr.press/v32/rezende14.html>.
- Risken, H. *The Fokker-Planck Equation*. Springer Berlin Heidelberg, 1984. doi: 10.1007/978-3-642-96807-5.
- Roweis, S. and Ghahramani, Z. Learning nonlinear dynamical systems using the expectation-maximization algorithm. In *Kalman Filtering and Neural Networks*, pp. 175–220. John Wiley & Sons, Inc., 2002. doi: 10.1002/0471221546.ch6.
- Rudy, S. H., Brunton, S. L., and Kutz, J. N. Smoothing and parameter estimation by soft-adherence to governing equations. *Journal of Computational Physics*, 398:108860, dec 2019. doi: 10.1016/j.jcp.2019.108860.
- Schmidt, D., Koppe, G., Monfared, Z., Beutelspacher, M., and Durstewitz, D. Identifying nonlinear dynamical systems with multiple time scales and long-range dependencies. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=_XYzwxPIQu6.
- Shi, Y., N. S., Paige, B., and Torr, P. Variational mixture-of-experts autoencoders for multi-modal deep generative models. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL <https://proceedings.neurips.cc/paper/2019/file/0ae775a8cb3b499ad1fc944e6f5c836-Paper.pdf>.
- Shi, Y., Paige, B., Torr, P., and N. S. Relating by contrasting: A data-efficient framework for multimodal generative models. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=vhKe9UFbrJo>.
- Smith, A. and Brown, E. Estimating a State-Space Model from Point Process Observations. *Neural Computation*, 15, 2003.
- Srivastava, N. and Salakhutdinov, R. R. Multimodal learning with deep boltzmann machines. In *Advances in Neural Information Processing Systems 25*, pp. 2222–2230. Curran Associates, Inc., 2012.
- Strogatz, S. H. *Nonlinear Dynamics and Chaos*. CRC Press, may 2018. doi: 10.1201/9780429492563.
- Sui, J., Adali, T., Yu, Q., Chen, J., and Calhoun, V. D. A review of multivariate methods for multimodal fusion of brain imaging data. *Journal of Neuroscience Methods*, 204(1):68–81, feb 2012. doi: 10.1016/j.jneumeth.2011.10.031.
- Sushko, I. and Gardini, L. Degenerate bifurcations and border collisions in piecewise smooth 1D and 2D maps. *International Journal of Bifurcation and Chaos*, 20(07):2045–2070, July 2010. doi: 10.1142/s0218127410026927. URL <https://doi.org/10.1142/s0218127410026927>.
- Sutter, T. M., Daunhawer, I., and Vogt, J. E. Generalized multi-modal ELBO. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=5Y21V0RDBV>.
- Trischler, A. P. and D’Eleuterio, G. M. Synthesis of recurrent neural networks for dynamical system simulation. *Neural Networks*, 80:67–78, aug 2016. doi: 10.1016/j.neunet.2016.04.001.
- Tsai, Y. H., Liang, P. P., Zadeh, A., Morency, L., and Salakhutdinov, R. Learning factorized multimodal representations. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL <https://openreview.net/forum?id=rygqqqsA9KX>.
- Turner, B. M., Forstmann, B. U., Wagenmakers, E.-J., Brown, S. D., Sederberg, P. B., and Steyvers, M. A bayesian framework for simultaneously modeling neural and behavioral data. *NeuroImage*, 72:193–206, may 2013. doi: 10.1016/j.neuroimage.2013.01.048.
- Vedantam, R., Fischer, I., Huang, J., and Murphy, K. Generative models of visually grounded imagination. In *International Conference on Learning Representations*, 2018. URL <https://openreview.net/forum?id=HkCsm6lRb>.
- 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. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2213):20170844, May 2018. doi: 10.1098/rspa.2017.0844. URL <https://doi.org/10.1098/rspa.2017.0844>.
- Wilson, H. R. *Spikes, Decisions, and Actions: The Dynamical Foundations of Neuroscience*. Oxford University Press, 1 edition, 1999.
- Woods, A. W. Turbulent plumes in nature. *Annual Review of Fluid Mechanics*, 42(1):391–412, jan 2010. doi: 10.1146/annurev-fluid-121108-145430.
- Wu, M. and Goodman, N. D. Multimodal generative models for scalable weakly-supervised learning. In *NeurIPS*, 2018.
- Yu, B. M., Afshar, A., Santhanam, G., Ryu, S. I., Shenoy, K. V., and Sahani, M. Extracting dynamical structure embedded in neural activity. In *Advances in Neural Information Processing Systems 18*, pp. 1545–1552. MIT Press, 2006.
- Zhao, Y. and Park, I. M. Variational online learning of neural dynamics. *Frontiers in Computational Neuroscience*, 14, October 2020. doi: 10.3389/fncom.2020.00071. URL <https://doi.org/10.3389/fncom.2020.00071>.

A. Methodological Details

A.1. EM algorithm: Optimization by Newton-Raphson Iterations

Given the distributional assumptions of the observation and latent models, the joint log-likelihood in (6) spells out as:

$$\begin{aligned}
 \log p_{\theta}(\mathbf{Z}, \mathbf{X}, \mathbf{C}) = & \\
 & -\frac{1}{2}(\mathbf{z}_1 - \boldsymbol{\mu}_0 - \mathbf{F}\mathbf{s}_1)^T \boldsymbol{\Sigma}^{-1}(\mathbf{z}_1 - \boldsymbol{\mu}_0 - \mathbf{F}\mathbf{s}_1) \\
 & -\frac{1}{2} \sum_{t=2}^T (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1} - \mathbf{W}\phi(\mathbf{z}_{t-1}) - \mathbf{h} - \mathbf{F}\mathbf{s}_t)^T \boldsymbol{\Sigma}^{-1}(\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1} - \mathbf{W}\phi(\mathbf{z}_{t-1}) - \mathbf{h} - \mathbf{F}\mathbf{s}_t) \\
 & -\frac{1}{2} \sum_{t=1}^T (\mathbf{x}_t - \mathbf{B}\phi(\mathbf{z}_t))^T \boldsymbol{\Gamma}^{-1}(\mathbf{x}_t - \mathbf{B}\phi(\mathbf{z}_t)) - \frac{T}{2}(\log |\boldsymbol{\Sigma}| + \log |\boldsymbol{\Gamma}|) \\
 & + \sum_{t=1}^T \left(\sum_{i=1}^{K-1} c_{it} \boldsymbol{\beta}_i \mathbf{z}_t - \log \left(1 + \sum_{j=1}^{K-1} \exp(\boldsymbol{\beta}_j \mathbf{z}_t) \right) \right) \\
 & + \text{const.}
 \end{aligned} \tag{10}$$

Let $\mathbf{z} = (z_{11}, \dots, z_{M1}, \dots, z_{1T}, \dots, z_{MT})^T \in \mathbb{R}^{MT \times 1}$ be the concatenated vector of all state variables across all time steps, $\mathbf{d}_{\Omega} := (d_{\Omega}^{(11)}, d_{\Omega}^{(21)}, \dots, d_{\Omega}^{(MT)})^T$ an indicator vector with $d_{\Omega}^{(mt)} = 1 \forall z_{mt} > 0$ and $d_{\Omega}^{(mt)} = 0$ otherwise, and $\mathbf{D}_{\Omega} := \text{diag}(\mathbf{d}_{\Omega})$. Arranging all terms quadratic, linear, and constant in \mathbf{z} into big matrix form (cf. (Koppe et al., 2019)), one can rewrite optimization criterion (6), (10) as

$$\begin{aligned}
 Q_{\Omega}^*(\mathbf{Z}) = & -\frac{1}{2} \left[2(\mathbf{a}^T \mathbf{a}) - 2\bar{\boldsymbol{\beta}}^T \mathbf{z} + \mathbf{z}^T \left(\mathbf{U}_0 + \mathbf{D}_{\Omega} \mathbf{U}_1 + \mathbf{U}_1^T \mathbf{D}_{\Omega} + \mathbf{D}_{\Omega} \mathbf{U}_2 \mathbf{D}_{\Omega} \right) \mathbf{z} \right] \\
 & -\frac{1}{2} \left[-\mathbf{z}^T (\mathbf{v}_0 + \mathbf{D}_{\Omega} \mathbf{v}_1) - (\mathbf{v}_0 + \mathbf{D}_{\Omega} \mathbf{v}_1)^T \mathbf{z} \right]
 \end{aligned} \tag{11}$$

where $\bar{\boldsymbol{\beta}} = (\bar{\boldsymbol{\beta}}_1, \dots, \bar{\boldsymbol{\beta}}_T) \in \mathbb{R}^{MT \times 1}$ is a column vector with vector elements $\bar{\boldsymbol{\beta}}_t := [\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_{K-1}, \mathbf{0}] \mathbf{c}_t \in \mathbb{R}^{M \times 1}$ (picking out the regression vector $\boldsymbol{\beta}_l$ associated with the selected category $c_{lt} = 1$ at time t), $\mathbf{a} = (\sqrt{\gamma_{z_1}}, \dots, \sqrt{\gamma_{z_T}})^T \in \mathbb{R}^{T \times 1}$ with $\gamma_{\mathbf{z}_t} := \log \left(1 + \sum_{j=1}^{K-1} \exp(\boldsymbol{\beta}_j^T \mathbf{z}_t) \right)$. The components of the \mathbf{U} -matrices and \mathbf{v} -vectors are defined in (Koppe et al., 2019) and restated here (with slight model-specific adjustments) for convenience. All matrices $\mathbf{U}_{\{0,1,2\}}$ have a block-tridiagonal structure as in Eq. 19 below with on- and off-diagonal blocks:

$$\mathbf{U}_0^{tt} = \boldsymbol{\Sigma}^{-1} + \mathbf{A}^T \boldsymbol{\Sigma}^{-1} \mathbf{A}, \quad \mathbf{U}_0^{t+1,t} = -\boldsymbol{\Sigma}^{-1} \mathbf{A}, \quad \mathbf{U}_0^{t,t+1} = (\mathbf{U}_0^{t+1,t})^T \quad \text{for } t = 1 \dots T-1, \quad \mathbf{U}_0^{T,T} = \boldsymbol{\Sigma}^{-1} \tag{12}$$

$$\mathbf{U}_1^{t,t} = \mathbf{W}^T \boldsymbol{\Sigma}^{-1} \mathbf{A}, \quad \mathbf{U}_1^{t,t+1} = -\mathbf{W}^T \boldsymbol{\Sigma}^{-1} \quad \text{for } t = 1 \dots T-1, \quad \mathbf{U}_1^{T,T} = \mathbf{0}_{M \times M} \tag{13}$$

$$\mathbf{U}_2^{t,t} = \mathbf{W}^T \boldsymbol{\Sigma}^{-1} \mathbf{W} + \mathbf{B}^T \boldsymbol{\Gamma}^{-1} \mathbf{B} \quad \text{for } t = 1 \dots T-1, \quad \mathbf{U}_2^{T,T} = \mathbf{B}^T \boldsymbol{\Gamma}^{-1} \mathbf{B} \tag{14}$$

The vectors $\mathbf{v}_{\{0,1\}}$ are given by:

$$\mathbf{v}_0 = [(\boldsymbol{\Sigma}^{-1} \mathbf{F} \mathbf{s}_1 - \mathbf{A}^T \boldsymbol{\Sigma}^{-1} (\mathbf{F} \mathbf{s}_2 + \mathbf{h}) + \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_0)^T, \dots, (\boldsymbol{\Sigma}^{-1} (\mathbf{F} \mathbf{s}_t + \mathbf{h}) - \mathbf{A}^T \boldsymbol{\Sigma}^{-1} (\mathbf{F} \mathbf{s}_{t+1} + \mathbf{h}))^T, \dots, (\boldsymbol{\Sigma}^{-1} (\mathbf{F} \mathbf{s}_T + \mathbf{h}))^T]^T \tag{15}$$

$$\mathbf{v}_1 = [(\mathbf{B}^T \boldsymbol{\Gamma}^{-1} \mathbf{x}_1 - \mathbf{W}^T \boldsymbol{\Sigma}^{-1} (\mathbf{F} \mathbf{s}_2 + \mathbf{h}))^T, \dots, (\mathbf{B}^T \boldsymbol{\Gamma}^{-1} \mathbf{x}_t - \mathbf{W}^T \boldsymbol{\Sigma}^{-1} (\mathbf{F} \mathbf{s}_{t+1} + \mathbf{h}))^T, \dots, (\mathbf{B}^T \boldsymbol{\Gamma}^{-1} \mathbf{x}_{T-1} - \mathbf{W}^T \boldsymbol{\Sigma}^{-1} (\mathbf{F} \mathbf{s}_T + \mathbf{h}))^T, (\mathbf{B}^T \boldsymbol{\Gamma}^{-1} \mathbf{x}_T)^T]^T \tag{16}$$

In the original formulation of the PLRNN algorithm (Durstewitz, 2017b; Koppe et al., 2019), the non-Gaussian terms due to $p(\mathbf{C} | \mathbf{Z})$ were lacking, and hence criterion Eq. (11) was piecewise quadratic (owing to the piecewise linear ReLU activation) and could be addressed by an efficient fixed-point-iteration algorithm that alternates between (i) solving the set of equations $dQ_{\Omega}^*(\mathbf{Z})/d\mathbf{Z} = 0$ linear in \mathbf{Z} and (ii) recomputing the ReLU-derivatives \mathbf{D}_{Ω} . Here we need to modify this algorithm, as the derivatives $dQ_{\Omega}^*(\mathbf{Z})/d\mathbf{Z}$ are not linear anymore, even for a fixed matrix \mathbf{D}_{Ω} . Luckily, however, Eq. (11) will still be piecewise concave (within each orthant of the objective function landscape) and hence we can efficiently solve it using the Newton-Raphson (NR) scheme

$$\mathbf{z}^{\text{new}} = \mathbf{z}^{\text{old}} - \alpha_{\mathbf{z}} \mathbf{H}^{-1} \nabla \mathbf{z}^{\text{old}} \tag{17}$$

where \mathbf{D}_Ω is recomputed after a few NR steps. Due to the Markov property of model Eq. 1, the Hessian $\mathbf{H} := \partial^2 Q_\Omega^*(\mathbf{Z}) / \partial \mathbf{z} \partial \mathbf{z}^T$ has a specific, block-tridiagonal structure (see Eq. (19) and Koppe et al. (2019)). This can be exploited (a) to store \mathbf{H} in sparse format and (b) to obtain the inverse in $\mathcal{O}(T)$ time (Paninski et al., 2009). The on- and off-diagonal blocks in \mathbf{H} are composed of the components specified above (i.e., in Eqn. (11) and below).

As noted in sect. 3.2, given the maximizer of Eq. (11) as an estimate of the mean and the Hessian \mathbf{H} , we can then compute the required expectations $E[\mathbf{z}]$, $E[\mathbf{z}\mathbf{z}^T]$, $E[\phi(\mathbf{z})]$, $E[\mathbf{z}\phi(\mathbf{z})^T]$ and $E[\phi(\mathbf{z})\phi(\mathbf{z})^T]$ mostly analytically (Durstewitz, 2017b; Koppe et al., 2019), based on which parameters $\theta_{\text{lat}} = \{\mu_0, \mathbf{A}, \mathbf{W}, \mathbf{F}, \mathbf{h}, \boldsymbol{\Sigma}\}$ and $\theta_X = \{\mathbf{B}, \boldsymbol{\Sigma}\}$ can be solved for analytically as a linear regression problem. For the parameters β in Eq. (8) this is not possible, but since the problem is still concave efficient NR updates can again be used to solve numerically by

$$\boldsymbol{\beta}^{\text{new}} = \boldsymbol{\beta}^{\text{old}} - \alpha_\beta \mathbf{J}^{-1} \nabla \boldsymbol{\beta}^{\text{old}}, \quad (18)$$

where $\nabla \boldsymbol{\beta}^{\text{old}} := \frac{\partial f(\boldsymbol{\theta})}{\partial \boldsymbol{\beta}}$ indicates the gradient, the Hessian is given by \mathbf{J} , and α_β is a learning rate (set to $\alpha_\beta = 0.001$ here). Using the analytical derivations and approximations outlined here and in sect. 3.2, both the E- and the M-step become reasonably fast.

A.2. Stepwise Training Protocol

It has been shown previously (Koppe et al., 2019) that the approximation of the true underlying DS is strongly improved by embedding the EM algorithm (described in section 3.2) into a stepwise training protocol that successively shifts the burden of reproducing the observations from the observation model $p_\theta(\mathbf{X}, \mathbf{C} | \mathbf{Z})$, as defined by Eqs. (2 & 3), to the latent process model $p_\theta(\mathbf{Z})$ as defined by Eq. (1). In a first step, a linear dynamical system (LDS) is trained by EM on the time series to find a suitable initial guess of parameters and states. Next, by deliberately fixing the covariance terms $\boldsymbol{\Gamma}$ and $\boldsymbol{\Sigma}$ of the observation and latent models, respectively, to certain values in the first full PLRNN runs, efficient training of the observation model is encouraged. In later steps, the observation model term $E[\log p_\theta(\mathbf{X}, \mathbf{C} | \mathbf{Z})]$ in optimization criterion Eq. (5) is clamped off completely, thus enforcing the temporal consistency requirements in Eq. (1) and hence forcing the latent dynamical model to capture the observed temporal evolution in its own prior dynamics $p_\theta(\mathbf{Z})$. We use the same strategy here. For further details please refer to (Koppe et al., 2019).

A.3. Parameterization Approximate Posterior

Owing to the latent model's Markov property, the $MT \times MT$ Hessian \mathbf{H} of the approximate posterior $q(\mathbf{Z} | \mathbf{X}, \mathbf{C})$ in Eq. (6) and Eq. (9) has a specific block-tridiagonal structure (see also (Paninski et al., 2009; Archer et al., 2015)):

$$\mathbf{H} = \begin{pmatrix} \mathbf{S}_1 & \mathbf{K}_1 & 0 & \cdots & \cdots & 0 \\ \mathbf{K}_1^T & \mathbf{S}_2 & \mathbf{K}_2 & 0 & \cdots & 0 \\ 0 & \mathbf{K}_2^T & \mathbf{S}_3 & \mathbf{K}_3 & 0 & \vdots \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & 0 & \mathbf{K}_{T-2}^T & \mathbf{S}_{T-1} & \mathbf{K}_{T-1} \\ 0 & 0 & 0 & 0 & \mathbf{K}_{T-1}^T & \mathbf{S}_T \end{pmatrix} \quad (19)$$

with $M \times M$ on-diagonal blocks \mathbf{S}_t and off-diagonal blocks \mathbf{K}_t .

For VI, we closely follow (Archer et al., 2015) who factorize the approximate posterior into a product of two Gaussians as

$$p_\theta(\mathbf{Z} | \mathbf{X}, \mathbf{C}) \approx q_\zeta(\mathbf{Z} | \mathbf{X}, \mathbf{C}) \propto q_1(\mathbf{Z} | \mathbf{X}, \mathbf{C}) q_0(\mathbf{Z}), \quad (20)$$

$$q_\zeta(\mathbf{Z} | \mathbf{X}, \mathbf{C}) = \mathcal{N}(\boldsymbol{\mu}_Z(\mathbf{X}, \mathbf{C}), \boldsymbol{\Lambda}_Z(\mathbf{X}, \mathbf{C})), \quad (21)$$

with

$$q_1(\mathbf{Z} | \mathbf{X}, \mathbf{C}) = \mathcal{N}(\mathbf{m}_{\zeta_\mu}, \mathbf{E}_{\zeta_\Lambda}), \quad (22)$$

$$q_0(\mathbf{Z}) = \mathcal{N}(0, \mathbf{D}). \quad (23)$$

Combining these two Gaussians yields for the combined mean $\boldsymbol{\mu}_Z$ and covariance $\boldsymbol{\Lambda}_Z$ of the posterior (Archer et al., 2015)

$$\boldsymbol{\Lambda}_Z(\mathbf{X}, \mathbf{C}) = (\mathbf{D}^{-1} + \mathbf{E}_{\zeta_\Lambda}^{-1}(\mathbf{X}, \mathbf{C}))^{-1}, \quad (24)$$

$$\boldsymbol{\mu}_Z(\mathbf{X}, \mathbf{C}) = \boldsymbol{\Lambda}_Z(\mathbf{X}, \mathbf{C}) \mathbf{E}_{\zeta_\Lambda}^{-1}(\mathbf{X}, \mathbf{C}) \mathbf{m}_{\zeta_\mu}(\mathbf{X}, \mathbf{C}), \quad (25)$$

where both \mathbf{D}^{-1} and $\mathbf{E}_{\zeta_\Lambda}^{-1}$ are $MT \times MT$ matrices of the general form given in Eq. (19). Because of this block-tri-diagonal form, matrix inversions can be done efficiently in $\mathcal{O}(T)$ time (Paninski et al., 2009).

The formulation in Eqs. (20-25) allows to insert a smoothness prior into the posterior via $q_0(\mathbf{Z})$. More specifically, the on-diagonal blocks \mathbf{P}_t and off-diagonal blocks \mathbf{K} of prior matrix \mathbf{D}^{-1} are time-independent and given by (tArcher15)

$$\mathbf{P}_1 = \mathbf{Q}_0^{-1} + \mathbf{V}^T \mathbf{Q}^{-1} \mathbf{V}, \quad (26)$$

$$\mathbf{P}_t = \mathbf{Q}^{-1} + \mathbf{V}^T \mathbf{Q}^{-1} \mathbf{V}, \quad t = 2, \dots, T-1, \quad (27)$$

$$\mathbf{P}_T = \mathbf{Q}^{-1}, \quad (28)$$

$$\mathbf{K} = -\mathbf{V}^T \mathbf{Q}^{-1}, \quad (29)$$

where $\mathbf{V} \in \mathbb{R}^{M \times M}$, $\mathbf{Q}_0 \in \mathbb{R}^{M \times M}$ and $\mathbf{Q} \in \mathbb{R}^{M \times M}$ are full parameter matrices optimized during training. This specific formulation follows the Kalman filter-smoother equations (Kalman, 1960; Paninski et al., 2009) where matrix \mathbf{D}^{-1} collects the covariance terms that come from the process model, and the specific form of Eqs. (26-29) ensures they are arranged in the correct way within the full Hessian Eq. (19). Matrix $\mathbf{E}_{\zeta_\Lambda}^{-1}$ in turn captures the time-dependent terms from the observation model which appear only in the blocks on the diagonal. These on-diagonal blocks \mathbf{L}_t as well as the time-dependent mean in Eq. (22) are parameterized through MLP-type neural network as

$$\mathbf{m}_{\zeta_\mu}^{(t)}(\mathbf{x}_t, \mathbf{c}_t) = \text{NN}_{\zeta_\mu}(\mathbf{x}_t, \mathbf{c}_t), \quad (30)$$

$$\mathbf{L}_t(\mathbf{x}_t, \mathbf{c}_t) = \text{NN}_{\zeta_\Lambda}(\mathbf{x}_t, \mathbf{c}_t), \quad (31)$$

with $\mathbf{m}_{\zeta_\mu}^{(t)} \in \mathbb{R}^{M \times 1}$ and $\mathbf{L}_t \in \mathbb{R}^{M \times M}$. The diagonal blocks in matrix Eq. (19) are then given by $\mathbf{S}_t = \mathbf{P}_t + \mathbf{L}_t$. In our case, each NN has two separate input layers for the two data modalities, followed by two hidden layers of dimension $d_h = 25$ each. The input and hidden layers of the respective data modality are fully connected. The third layer combines the two input streams (therefore has input dimension $d_c = 2 \cdot d_h = 50$), followed by an additional hidden layer, again fully connected. For all but the output layer we use ReLU activation functions.

The VI code was implemented in Python/PyTorch, while for EM we modified a previous MatLab (MathWorks Inc.) implementation (Koppe et al., 2019). All experiments were run on CPU-based servers (Intel Xeon Plat 8160 @ 2.1GHz with 24 cores or Intel Xeon Gold 6148 @ 2.4GHz with 20 cores), and the EM and VI algorithms were comparable in runtime (around 400 minutes on a single core, i.e. without parallelization, for training on one Lorenz data set as used in sects. 4.1 and B.6).

A.4. Hyper-parameter Selection

Expectation-Maximization In general, the EM algorithm does not have many hyper-parameters that are critical to tune. The most crucial one is the number of latent states M , which we checked for the Lorenz benchmarks within the range $M \in \{12, 15, 16, 18\}$ based on previous work (Koppe et al., 2019). $M = 15$ was selected as it led to the best state space reconstructions as assessed by our Kullback-Leibler measure D_{KL} (see sect. B.2) averaged across 20 trajectories. Furthermore, a grid search was performed to determine the optimal Newton-Raphson learning rates in Eqn. 17 and 18 across the ranges $\alpha_z = \{0.6, 0.8, 1\}$ (E-step) and $\alpha_\beta = \{0.0005, 0.001, 0.005, 0.01\}$ (M-step), respectively. To strike a balance between good model fit, assessed by the model's ELBO approximation, and runtime (given the many samples drawn to create the cumulative histograms), values were chosen at which the ELBO did not notably increase anymore yet runtime was comparatively low.

For the fMRI data, we again followed Koppe et al. (2019) for orientation and searched for an optimal number of latent states within $M \in \{15, 20, 25, 30\}$. $M = 20$ was selected as it yielded the lowest $n = 10$ step ahead prediction error (cf. Fig. 3A). Similar as for the Lorenz benchmarks, a grid search for the optimal Newton-Raphson learning rates was carried out within $\alpha_z \in \{0.6, 0.8, 1\}$ (E-step) and $\alpha_\beta \in \{0.0001, 0.00025, 0.0005, 0.00075, 0.001\}$ (M-step; for $\alpha_\beta > 0.001$ NR steps did not converge). As for the Lorenz benchmarks, parameters were selected that achieved a compromise between a reasonably high ELBO (without much further improvement for lower rates) and runtime (given the many runs across subjects and left-out sets that needed to be performed). In general, models were trained starting from five different initial conditions for the parameters, and from those retaining the configuration which yielded the highest likelihood.

Variational Inference For comparability with the EM algorithm, the number of latent dimensions M was fixed to be the same for VI. For the encoder network we explored a number of different settings with respect to performance and runtime: number of hidden layers $n_h \in \{2, 3, 4, 5\}$, number of units per layer $d_h \in \{10, 15, 25, 50, 75, 100\}$, learning rate $\alpha_v \in \{0.002, 0.001, 0.01\}$, as well as multi-layer perceptron (MLP) and convolutional neural network (CNN) architectures. Keeping runtime at a minimum and still obtaining good performance resulted in the parameter settings $n_h = 2$, $d_h = 25$, $\alpha_v = 0.001$ and the MLP architecture. For the multimodal setup, an additional concatenation layer followed by another hidden layer was used.

LSTM Hyper-parameters of the LSTM were chosen according to the lowest average MSE across n -step ahead predictions over the range $n \in \{5, 7, 9, 11, 13, 15\}$. For comparability, a similar range for the number of LSTM cell states was explored as for the PLRNN's latent space dimensionality, $M = \{15, 17, 20, 21, 23, 25\}$, for which $M = 20$ turned out to yield the best performance. For all other hyper-parameters, the ranges explored followed Hochreiter et al. (2007). Specifically, optimal learning rates and the number of training epochs were determined by probing the ranges $\lambda \in \{0.001, 0.002, 0.003, 0.004, 0.005, 0.006, 0.007, 0.008, 0.009, 0.01\}$ and $\#\text{epochs} \in \{25, 50, 75, 100, 125, 150, 175, 200\}$, respectively.

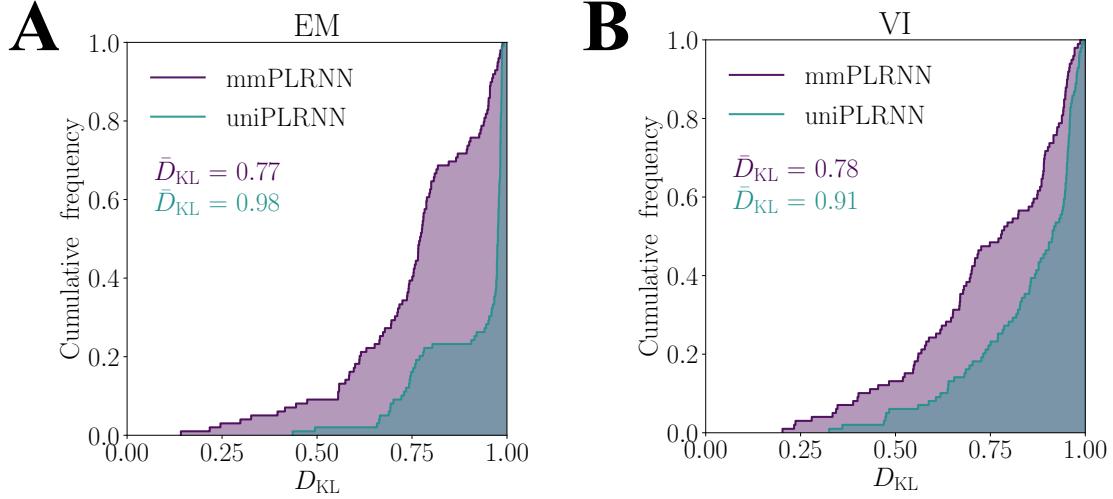


Figure 5: Same setup as in Fig. 2, but with one Lorenz variable (x_2) omitted from the observations. A) Cumulative performance histograms ($n = 100$ runs) as in Fig. 2C for continuous+categorical (purple) vs. only (degenerated) continuous (cyan) information channel for models trained by EM. \bar{D}_{KL} indicates the median. B) Same as A for models trained by VI.

B. Details on Experimental Setups and Performance Measures

B.1. Lorenz Equations

The (stochastic) 3D-Lorenz system (Lorenz, 1963) is defined by the set of equations

$$\begin{aligned} dx_1 &= s(x_2 - x_1)dt + d\epsilon_1(t) \\ dx_2 &= (x_1(r - x_3) - x_2)dt + d\epsilon_2(t) \\ dx_3 &= (x_1x_2 - bx_3)dt + d\epsilon_3(t). \end{aligned} \quad (32)$$

The system was solved with 4th order Runge-Kutta numerical integration. Process noise was injected by adding an *i.i.d.* Gaussian term $d\epsilon \sim \mathcal{N}(0, 0.0025 \times dt \mathbf{I})$ to the three equations. Parameter values used here were $s = 10$, $r = 28$, and $b = 8/3$, which place the Lorenz system into the chaotic regime.

B.2. Agreement in Attractor Geometries

Following Koppe et al. (2019) and Schmidt et al. (2021), we quantify the agreement in attractor geometries by comparing the true and model-generated probability distributions across observations \mathbf{X} in state space through a Kullback-Leibler divergence (D_{KL}), approximated as

$$D_{KL}(\hat{p}_{\text{true}}^{(k)}(\mathbf{x}) \parallel \hat{p}_{\text{gen}}^{(k)}(\mathbf{x} \mid \mathbf{z})) \approx \sum_{k=1}^K \hat{p}_{\text{true}}^{(k)}(\mathbf{x}) \log \left(\frac{\hat{p}_{\text{true}}^{(k)}(\mathbf{x})}{\hat{p}_{\text{gen}}^{(k)}(\mathbf{x} \mid \mathbf{z})} \right) \quad (33)$$

where $\hat{p}_{\text{true}}(\mathbf{x})$ is the true distribution of observations across state space, $\hat{p}_{\text{gen}}(\mathbf{x} \mid \mathbf{z})$ the simulated distribution generated by the (freely running) PLRNN, and index k runs across bins in state space. See Koppe et al. (2019) or Schmidt et al. (2021) for more details. To evaluate D_{KL} , trajectories of length $T = 100\,000$ were generated from both ground truth system and PLRNN, from which the initial transients (500 time points) were cut off. To yield a relative measure in $[0, 1]$, D_{KL} was normalized to the largest $D_{KL}^{\max} \approx 18.4$ across all iterations from both the noisy and incomplete Lorenz experiments using either EM or VI (i.e., same maximum value was used for all graphs in Figs. 2, 5).

In order to compute D_{KL} in observation space for the case where one Lorenz variable was missing (see below), a projection from the latent into the *full* observation space was computed by linear regression (i.e., re-computing matrix \mathbf{B} from observation model Eq. (2) for the full set of observations).

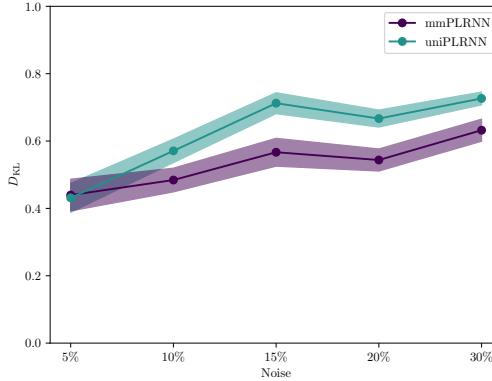


Figure 6: Mean D_{KL} and SEM ($n = 40$) for the uni- vs. mmPLRNN trained by VI on the noisy Lorenz benchmark across longer time series of $T = 2500$ for different observation noise levels. Note that a) performance for VI at 10% improved to levels comparable to EM for these longer time series as compared to the $T = 1000$ series used in the main text (cf. Fig. 2), and b) access to additional categorical information significantly improves reconstruction performance only as the continuous-Gaussian channel becomes more severely corrupted by noise, but not for low noise levels (5%), as one may intuitively expect.

B.3. Details on fMRI Experiments and Analysis

Briefly, human study participants (Koppe et al., 2014) were presented with a sequence of images of different shapes (rectangles and triangles) under three different task conditions while lying in a fMRI scanner: The continuous delayed response 1-back task (CDRT), the continuous matching 1-back task (CMT), and a 0-back control choice reaction task (CRT). In all three task stages subjects had to correctly indicate the type of stimulus currently presented (0-back) or 1 step before in the sequence (1-back). Task blocks were presented sequentially and repeated 5 times (amounting to 3×5 task blocks), and only differed w.r.t. the instruction phase and displayed response options. In addition to these three task phases, a resting condition where the participants were just lying still in the scanner with eyes closed, and an instruction phase which informed the participants about the upcoming task phase, were included in the experiments. These constituted the five task stages, each of which involving different mental processes, which were assigned different categorical labels for decoding. Any external information concerning the type of stimulus presented was omitted during training, in order to not provide the algorithm with any other source of information about the labels or dynamics. Analysis of BOLD signals was performed on the first principal components of 10 brain regions bilaterally relevant to the n-back task (Owen et al., 2005) (yielding $N = 20$ time series per subject). The details of the experimental setup are given in (Koppe et al., 2014); fully anonymized data were obtained from the authors of that study and used here with their permission.

The confusion matrix reported in Fig. 3C was determined through 5-fold cross-validation. Specifically, this was done by fixing the mmPLRNN's parameters from the training set and obtaining posterior state estimates $E[\mathbf{Z}^{\text{test}} | \mathbf{X}^{\text{test}}]$ from the left-out BOLD signal \mathbf{X}^{test} alone (using the pre-trained encoder model), after re-estimating the initial condition μ_0 on the left-out segment. These inferred latent trajectories were then used to predict the unseen task labels through the previously trained observation model $p_{\theta_{\text{cat}}}(\mathbf{C}^{\text{test}} | \mathbf{Z}^{\text{test}})$ (Eq. (3)). Only validation blocks from all subjects were included for which the BOLD dynamics was reconstructed successfully on the respective training set (only in these cases the training was considered successful; note that the quality of the recordings may differ considerably among subjects). This yielded a total of $k = 84$ left-out test sets from $l = 21$ subjects. Relative frequencies were then computed by first summing across all these test sets from those subjects and then dividing by the respective total counts.

For all analyses in Figs. 3 & 4, data from $l = 21$ subjects were used, excluding 5 subjects from a total of 26 due to apparent artefacts in the BOLD signal or model divergence (including subjects with artefacts did, however, not change the results of the analyses, i.e., significant non-/differences remained as reported in the graph).

For the analysis of class label prediction (Fig. 3C), all classifiers received the same 20 BOLD principal components as inputs and the 5 task class labels as to-be-predicted outputs as used for mmPLRNN training. For SVM, radial basis function (RBF) kernels with tradeoff parameter $C = 1.0$ were used. All classifiers were trained using the scikit-learn library.

B.4. MSE n -step ahead Prediction

We define the MSE for n -step ahead predictions as

$$\text{MSE}(n) = \frac{1}{N \cdot (T - n)} \sum_{t=1}^{T-n} \|\mathbf{x}_{t+n} - \hat{\mathbf{x}}_{t+n}\|_2^2 \quad (34)$$

where $\hat{\mathbf{x}}_{t+n} \in \mathbb{R}^{N \times 1}$ is produced by iterating model Eq. (1) n steps forward in time from its current best estimator $E[\mathbf{z}_t | \mathbf{x}_{1:T}]$, and generating from this forward-iterated value $\hat{\mathbf{z}}_{t+n}$ the prediction $\hat{\mathbf{x}}_{t+n} := E[\mathbf{x}_{t+n} | \hat{\mathbf{z}}_{t+n}]$ according to observation model Eq. (2).

B.5. Calculation of Lyapunov Exponents

Let us denote the PLRNN mean function as given in Eq. 1 by $F_\theta(\mathbf{z}_{t-1}, \mathbf{s}_t)$. Then the system's Jacobian at time t is given by

$$\mathbf{J}_t := \frac{\partial F_\theta(\mathbf{z}_{t-1}, \mathbf{s}_t)}{\partial \mathbf{z}_{t-1}} = \mathbf{A} + \mathbf{W} \mathbf{D}_{\Omega(t-1)}, \quad (35)$$

where $\mathbf{D}_{\Omega(t-1)}$ is a diagonal $\{0, 1\}$ matrix with the *ReLU* derivatives at time point $t-1$ on its diagonal, i.e. an $M \times M$ submatrix of \mathbf{D}_Ω as defined in Appx. A.1. The maximal Lyapunov exponent along a PLRNN trajectory $\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_T, \dots\}$ is then defined as

$$\lambda_{max} := \lim_{T \rightarrow \infty} \frac{1}{T} \log \left\| \prod_{r=0}^{T-2} \mathbf{J}_{T-r} \right\|, \quad (36)$$

where $\|\cdot\|$ denotes the spectral norm (or any subordinate norm) of a matrix. Practically, one can let the orbit evolve until λ_{max} in Eq. 36 converges for some T .

B.6. Benchmarks: Lorenz System with Incomplete Gaussian and Categorical Observations

As a second test case, we studied whether additional categorical information (as in sect. 4.1, Fig. 2) could also help to identify the chaotic Lorenz system when one of its dynamical variables (x_2) was missing from the observations, i.e. only $\mathbf{x}_t^{\text{red}} = (x_{1t}, x_{3t})^T$ was provided. This is a nontrivial case, since the Lorenz system is a highly condensed minimal model for the chaotic attractor dynamics, i.e. with each variable absolutely necessary to produce the observed behavior (unlike many experimental systems which often have quite some redundancy, as in the nervous system or molecular networks). Yet, as shown in Fig. 5, non-quantitative, categorical data could efficiently compensate for the lack of continuous time series information about one of the system's variables. In terms of summary statistics, this is reflected in the D_{KL} distributions (Fig. 5) when the mmPLRNN inference was run with (purple) vs. without (cyan) access to categorical data on top of the linearly transformed (x_{1t}, x_{3t}) time series. Again this was generally true for both EM and VI, with EM performing somewhat better on average.

C. Examples of GLM-type Observation Models: Categorical, Gamma and Zero-Inflated Poisson Distributions

For the categorical model, Eq. 3, that we explored in the main text, the natural link function is given by

$$\begin{aligned} \pi_i &= \frac{\exp(\boldsymbol{\beta}_i^T \mathbf{z}_t)}{1 + \sum_{j=1}^{K-1} \exp(\boldsymbol{\beta}_j^T \mathbf{z}_t)} \in [0, 1] \quad \forall i \in \{1 \dots K-1\} \\ \pi_K &= \frac{1}{1 + \sum_{j=1}^{K-1} \exp(\boldsymbol{\beta}_j^T \mathbf{z}_t)}, \text{ such that } \sum_{i=1}^K \pi_i = 1, \end{aligned} \quad (37)$$

where $\boldsymbol{\beta}_i \in \mathbb{R}^{M \times 1}$ is the vector of regression weights for category $i = 1 \dots K-1$.

Here we illustrate the VI-based mmPLRNN on two further examples of observation models, namely when we have observations that could best be accounted for by 1) a gamma-distribution or by 2) a Poisson distribution with an excess of zeros (Zero-Inflated Poisson (ZIP) model (Lambert, 1992)). Examples of the latter are event counts for earthquakes or a neuron's action potentials where occasional periods of increased activity may be separated by relatively long periods of silence.

Real-valued gamma time series $\mathbf{G} = \{\mathbf{g}_t\}$, $\mathbf{g}_t \in \mathbb{R}_+^{J \times 1}$, $t = 1 \dots T$, would be described by the conditional density

$$\mathbf{g}_t \mid \mathbf{z}_t \sim \text{Gamma}(\omega, \boldsymbol{\nu}_t) \quad (38)$$

where $\omega > 0$ is a shape parameter and $\boldsymbol{\nu}_t \in \mathbb{R}_+^{J \times 1}$ are scale parameters. We may connect them to the latent states \mathbf{z}_t in model Eq. (1) through a GLM, where we model the distribution's conditional means $\boldsymbol{\mu}_t = (\mu_{1t}, \dots, \mu_{Jt})^T = \omega / \boldsymbol{\nu}_t$ at time t via the log link function:

$$\log \mu_{jt} = \boldsymbol{\xi}_j^T \mathbf{z}_t \quad \forall j \in \{1 \dots J\} \quad (39)$$

where $\boldsymbol{\xi}_j \in \mathbb{R}^{M \times 1}$ is a vector of regression weights for each of the gamma observations $j = 1 \dots J$. Note that like for the Gaussian and categorical models we did not include the usual constant offset (bias) term in the GLM, as we assume the overall level is determined by the latent states \mathbf{z}_t which are equipped with their own bias terms \mathbf{h} (cf. Eq. (1), avoiding model redundancy).

As an example of a somewhat more complex, composite distributional model, assume we have integer-valued Poisson data $\mathbf{Q} = \{\mathbf{q}_t\}$, $\mathbf{q}_t \in \mathbb{N}^{L \times 1}$, $t = 1 \dots T$, but with an excess proportion of zeros. This situation could be described by the ZIP model (Lambert, 1992)

which assumes that each observation q_{lt} at time t is either 0 with probability ψ_{lt} , or distributed according to a Poisson process with mean λ_{lt} with probability $1 - \psi_{lt}$:

$$\mathbf{q}_t \mid \mathbf{z}_t \sim \text{ZIP}(\boldsymbol{\psi}_t, \boldsymbol{\lambda}_t) \quad (40)$$

$$p(q_{lt} \mid \mathbf{z}_t) = \begin{cases} \psi_{lt} + (1 - \psi_{lt})e^{-\lambda_{lt}} & \text{for } q_{lt} = 0 \\ (1 - \psi_{lt}) \frac{\lambda_{lt}^{q_{lt}}}{q_{lt}!} e^{-\lambda_{lt}} & \text{for } q_{lt} > 0 \end{cases} \quad (41)$$

The elements of probability vector $\boldsymbol{\psi}_t = (\psi_{1t}, \dots, \psi_{Lt})^T$ are produced from the latent states \mathbf{z}_t through the logit link function

$$\log \frac{\psi_{lt}}{1 - \psi_{lt}} = \boldsymbol{\eta}_l^T \mathbf{z}_t \quad \forall l \in \{1 \dots L\}, \quad (42)$$

where $\boldsymbol{\eta}_l \in \mathbb{R}^{M \times 1}$ is a vector of regression weights for each Poissonian variable $l = 1 \dots L$. Likewise, the mean values $\boldsymbol{\lambda}_t = (\lambda_{1t}, \dots, \lambda_{Lt})^T$ of the Poisson distribution are connected to the latent states \mathbf{z}_t through the log-link function

$$\log \lambda_{lt} = \boldsymbol{\gamma}_l^T \mathbf{z}_t \quad \forall l \in \{1 \dots L\}, \quad (43)$$

where $\boldsymbol{\gamma}_l \in \mathbb{R}^{M \times 1}$ is another vector of regression weights for each of the Poisson observations $l = 1 \dots L$.

Assuming, as for the Gaussian and categorical observations, that the gamma (\mathbf{g}_t) and ZIP (\mathbf{q}_t) observations are conditionally independent given the latent state \mathbf{z}_t , the log-likelihood for this setup including Gaussian data is given by the following factorization:

$$\begin{aligned} p_\theta(\mathbf{x}, \mathbf{g}, \mathbf{q}) &= \int_{\mathbf{Z}} p_\theta(\mathbf{z}_1) \prod_{t=2}^T p_\theta(\mathbf{z}_t \mid \mathbf{z}_{t-1}) \prod_{t=1}^T p_\theta(\mathbf{x}_t \mid \mathbf{z}_t) \\ &\quad \prod_{t=1}^T p_\theta(\mathbf{q}_t \mid \mathbf{z}_t) \prod_{t=1}^T p_\theta(\mathbf{g}_t \mid \mathbf{z}_t) d\mathbf{Z} \end{aligned} \quad (44)$$

with parameters $\boldsymbol{\theta} = \{\boldsymbol{\mu}_0, \mathbf{A}, \mathbf{W}, \mathbf{F}, \mathbf{h}, \boldsymbol{\Sigma}, \mathbf{B}, \boldsymbol{\Gamma}, \omega, \{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_J\}, \{\boldsymbol{\eta}_1, \dots, \boldsymbol{\eta}_L\}, \{\boldsymbol{\gamma}_1, \dots, \boldsymbol{\gamma}_L\}\}$. Again we approximate this by the ELBO as given for categorical+Gaussian observations in Eq. (5), and parameterize the variational approximation $q_\zeta(\mathbf{Z} \mid \mathbf{X}, \mathbf{G}, \mathbf{Q})$ through neural networks in the very same way as described in Appx. A.3 above.

Fig. 7 illustrates the application to the noisy Lorenz setting (as described in sect. 4.1), this time with gamma and/or ZIP observations on top, or instead, of Gaussian observations. Simulations were run with $M = 15$ latent states, a linear-Gaussian observation model in addition to the two models described above, and 20% observation noise in the Gaussian channel. For the mmPLRNN including all 3 modalities, the NN parameterizing the approximate posterior had three separate input layers for the three data modalities, followed by two hidden layers of dimension $d_h = 25$ each. A third layer of dimension $d_c = 75 = 3d_h$, where 3 is the number of modalities, combined the modality-specific streams, followed by two additional hidden layers of size d_h . All layers were fully connected with ReLU activation, except for the output layer. As the graphs indicate, additional observations from other modalities again help to reconstruct the attractor geometry if the Gaussian observations are very noisy. We also observed, however, that it is difficult to reconstruct the Lorenz system from (by definition non-negative) gamma or ZIP observations alone. At least for ZIP-distributed data this is easy to explain (see example time series in Fig. 7B), as these - by model definition - low and sparse counts provide only very little information about the Lorenz attractor's complex geometry.

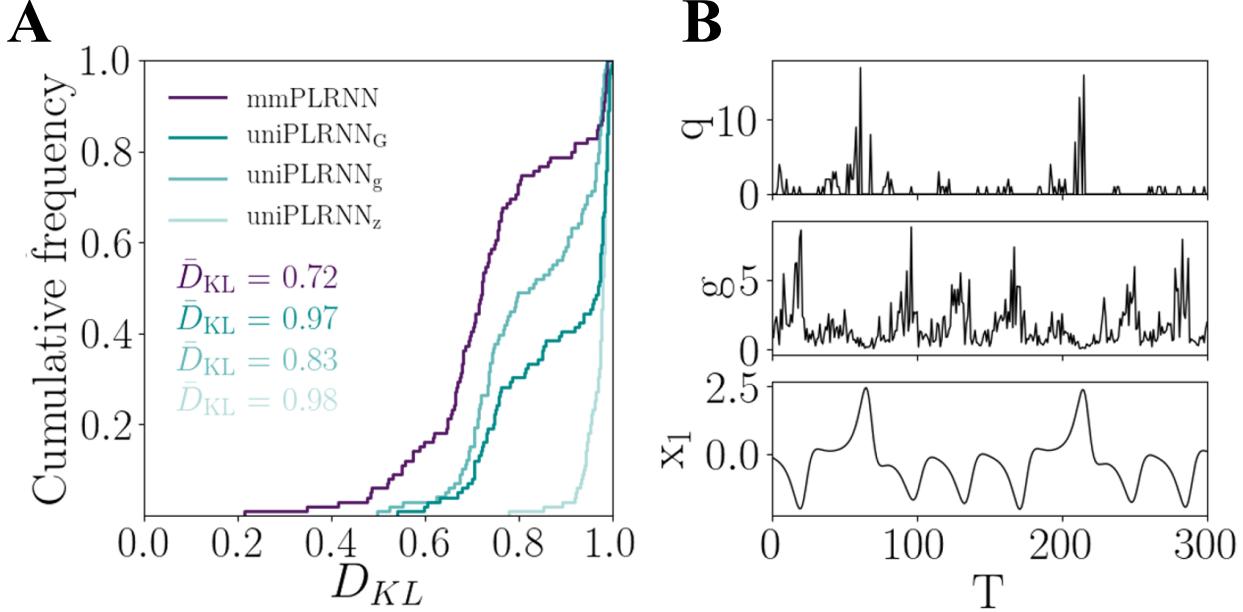


Figure 7: A) Cumulative histograms ($n = 100$ runs) of D_{KL}/D_{KL}^{max} for all three (Gaussian, gamma, ZIP) uni-modal PLRNNs (from light to dark cyan) and the mmPLRNN connected to all three observation modalities simultaneously (purple). \bar{D}_{KL} indicates the median. G = Gaussian, g = gamma, Z = ZIP model. B) Example time series of ZIP (top) and gamma (center) observations generated from the Lorenz attractor dynamics for which the time series of one variable is shown at the bottom. Note that the original Lorenz time series information appears highly degraded in the sparse ZIP output, and the structure is also distorted in the gamma output.

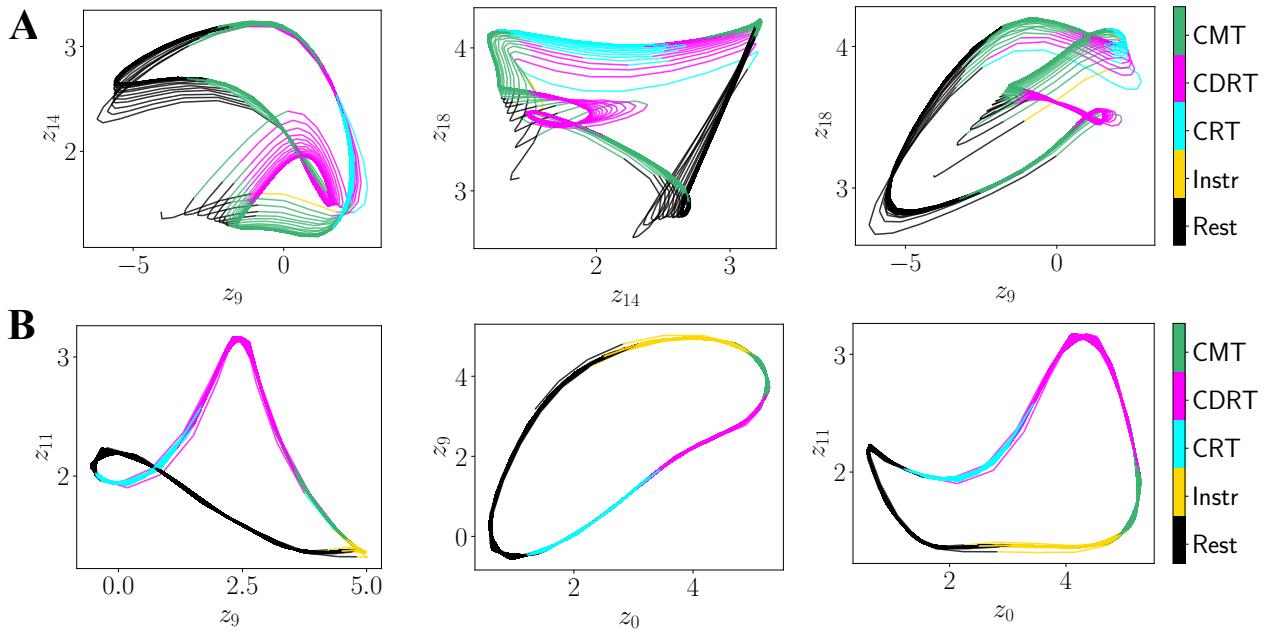


Figure 8: Two further examples for the association between predicted class labels (color coding) and learned BOLD dynamics. Shown are 2d subspaces of a mmPLRNN's generated state space. Subspaces chosen for display were selected according to Fisher's discriminant criterion.