InfoDPCCA: Information-Theoretic Dynamic Probabilistic Canonical Correlation Analysis

Shiqin Tang[1,2] **Shujian Yu**[3,4]

[1]Dept. of Data Science , City University of Hong Kong, Hong Kong
[2]Inst. of Artificial Intelligence , Peking University, Beijing, China
[3]Dept. of Computer Science , Vrije Universiteit Amsterdam, Amsterdam, Netherlands
[4]Dept. of Physics and Technology , UiT The Arctic University of Norway, Tromsø, Norway

Abstract

Extracting meaningful latent representations from high-dimensional sequential data is a crucial challenge in machine learning, with applications spanning natural science and engineering. We introduce InfoDPCCA, a dynamic probabilistic Canonical Correlation Analysis (CCA) framework designed to model two interdependent sequences of observations. InfoDPCCA leverages a novel information-theoretic objective to extract a shared latent representation that captures the mutual structure between the data streams and balances representation compression and predictive sufficiency while also learning separate latent components that encode information specific to each sequence. Unlike prior dynamic CCA models, such as DPCCA, our approach explicitly enforces the shared latent space to encode only the mutual information between the sequences, improving interpretability and robustness. We further introduce a two-step training scheme to bridge the gap between information-theoretic representation learning and generative modeling, along with a residual connection mechanism to enhance training stability. Through experiments on synthetic and medical fMRI data, we demonstrate that InfoDPCCA excels as a tool for representation learning. Code of InfoDPCCA is available at `https://github.com/marcusstang/InfoDPCCA`.

1 INTRODUCTION

Extracting meaningful latent representations from complex, high-dimensional sequential data is a fundamental challenge in machine learning, with broad applications across natural science and engineering. For instance, state-space models have been employed to uncover dynamic patterns in brain activity, aiding in the diagnosis of psychiatric disorders [Suk et al., 2016], and to track functional connectivity changes in resting-state fMRI for cognitive impairment assessment [Kusano et al., 2019].

In this paper, we introduce InfoDPCCA, a dynamic probabilistic Canonical Correlation Analysis (CCA) framework designed to model two interdependent sequences of observations. Our approach extracts a shared latent representation that captures the mutual information between the two data streams while simultaneously learning distinct components unique to each sequence. By leveraging a novel information-theoretic objective and a two-step training scheme, InfoDPCCA encourages the shared latent states to encode only the mutual information between the observations and strike a balance between representation compression and predictive sufficiency. InfoDPCCA can be viewed both as a generative model and a stochastic representation learning method.

The paper is organized as follows: In Section 2, we review the recent development of CCA and the usage of Information Bottleneck in sequential modeling. Section 3 describes simpler probabilistic models that form the backbones of our approach. In Section 4.1, we introduce Deep Dynamic Probabilistic CCA (D^2PCCA) [Tang et al., 2025b] as a baseline for comparison. Section 4.2 compares the graphical models of D^2PCCA and InfoDPCCA and provides justifications for the change in design choice. Section 4.3 identifies a key limitation of D^2PCCA and introduces an information-theoretic objective to mitigate this issue. Section 4.4 outlines a two-step training scheme for InfoDPCCA. Section 4.5 proposes methods to improve the stability and efficiency of the training process. Finally, in Section 5, we provide numerical simulations and real-world medical fMRI data to validate that InfoDPCCA fulfills its claims and excels as a tool for representation learning.

The major contributions of the paper include:

1. The proposal of a novel dynamic CCA model that improves upon its predecessor D^2PCCA.

Accepted for the 41st Conference on Uncertainty in Artificial Intelligence (UAI 2025).

2. A novel information-theoretic objective to extract shared latent states as well as its tractable variational bound.
3. A two-step training approach that bridge the gap between information-theoretic representation learning and generative modeling.
4. A novel residual connection proposed in Section 4.5 to enhance the training stability.

2 RELATED WORKS

The method of using Information Bottleneck (IB) [Tishby et al., 2000] to guide the training of sequential models is well-established. Chechik et al. [2005] proposes Gaussian IB (GIB) that extends the IB principle to multivariate Gaussian variables, showing that GIB's optimal projections align with CCA's eigenstructure. Creutzig et al. [2009] and Amir et al. [2015] apply IB to linear dynamical systems, focusing on model reduction and system realization. Kolchinsky et al. [2019] extends IB to nonlinear encoding and decoding maps and proposes a non-parametric upper bound on the mutual information (MI) term. Fischer [2020] discusses the possibility of extending its static IB framework to sequence learning. Meng et al. [2022] maximizes predictive information in the representation space and adapts InfoNCE's [Oord et al., 2018] MI lower bound to achieve a low-variance approximation. Kalajdzievski et al. [2022] leverages IB to learn compressed latent representations while preserving relevant transfer entropy information, which quantifies the causal influence of a source process on the future states of a target process. Murphy et al. [2024] further decomposes transfer entropy into its contributions from the source's past and the target's future using IB. Federici et al. [2023] applies IB to learn latent representations that retain essential system dynamics while discarding superfluous information, such as high-frequency noise, irrelevant features, and short-term dependencies. Another notable extension is the Multi-view Information Bottleneck [Federici et al., 2020], which adapts IB to learn the representations of each view of the data that capture the shared information across multiple views.

CCA is a statistical method that explores the shared structure of paired datasets by finding linear projections that maximize their correlation. Painsky et al. [2025] extends CCA to nonlinear settings, making the latent representations more flexible and expressive. Motivated by the intuition behind CCA, Bach and Jordan [2005] and Klami and Kaski [2008] propose probabilistic versions of CCA by modeling an underlying shared latent state that generates both observations. Wang et al. [2016] further extends probabilistic CCA to nonlinear observation models using amortized variational inference [Kingma and Welling, 2013]. Dynamic Probabilistic CCA (DPCCA) [Nicolaou et al., 2014] is a dynamic extension of CCA that models the evolution of two variables simultaneously over time. Tang et al. [2025b] builds upon DPCCA by introducing nonlinear state transitions and emission models, making it more expressive for complex time-series data. We refer to both linear and nonlinear versions of the model as DPCCA.

3 BACKGROUND

In this section, we briefly introduce the static Variational Information Bottleneck (VIB) [Alemi et al., 2016] and Conditional Entropy Bottleneck (CEB) [Fischer, 2020] models and propose a dynamic extension of VIB, which we name Dynamic VIB (DVIB). These models serve as the foundation for extending our approach to the more complex case of two observations.

Information Bottleneck (IB) [Tishby et al., 2000] is a framework that seeks to optimize the trade-off between compressing the input $\mathbf{x}$ into a compact representation $\mathbf{z}$ while preserving relevant information about the target $\mathbf{y}$, with the goal of minimizing the mutual information between $\mathbf{x}$ and $\mathbf{z}$, $I(\mathbf{x};\mathbf{z})$, while maximizing the mutual information between $\mathbf{z}$ and $\mathbf{y}$, $I(\mathbf{z};\mathbf{y})$, i.e.

$$\min \quad \beta I(\mathbf{z};\mathbf{x}) - I(\mathbf{z};\mathbf{y}), \tag{1}$$

where β is a hyperparameter controlling the contribution of the two terms.

VIB models the Markov chain $\mathbf{y} \leftrightarrow \mathbf{x} \leftrightarrow \mathbf{z}$, where $\mathbf{z}$ is a stochastic latent representation of $\mathbf{x}$. VIB considers an additional observed variable $\mathbf{y}$ such that the three variables follow the joint distribution $p(\mathbf{x},\mathbf{y},\mathbf{z}) = p(\mathbf{x})p(\mathbf{y}|\mathbf{x})q_\phi(\mathbf{z}|\mathbf{x})$, where ϕ parameterizes the stochastic encoder. VIB optimizes the IB (1) by replacing the mutual information terms with tractable variational bounds.

On the other hand, Fischer [2020] optimizes the CEB objective,

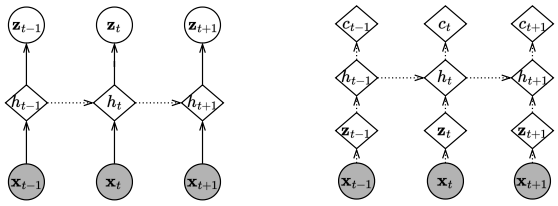
$$\min \quad I(\mathbf{z};\mathbf{x}|\mathbf{y}) - \lambda I(\mathbf{z};\mathbf{y}), \tag{2}$$

where the conditional mutual information (CMI) is given by

$$I(\mathbf{z},\mathbf{x}|\mathbf{y}) = I(\mathbf{x};\mathbf{z}) - I(\mathbf{z};\mathbf{y}). \tag{3}$$

The CEB objective can be rewritten in the same form as (1) since (2) can be expressed as $I(\mathbf{z};\mathbf{x}) - (1+\lambda)I(\mathbf{z};\mathbf{y})$. However, unlike IB, the two terms in the CEB objective are disjoint, allowing CEB to achieve a tighter approximation through a better choice of the inference network. Furthermore, by representing the IB as CEB, we can track the evolution of the conditional mutual information term, which can be driven to zero in an optimal case.

To obtain a lower bound on $I(\mathbf{z};\mathbf{y})$, both VIB and CEB require training a classifier network $p_\theta(\mathbf{y}|\mathbf{z})$. Once training is complete, classification is performed as $p(\mathbf{y}|\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[p_\theta(\mathbf{y}|\mathbf{z})]$. This contrasts with the Conditional



(a) DVIB (b) InfoNCE

Figure 1: Graphical models for DVIB and InfoNCE. The shaded nodes denote observations, while the unshaded nodes denote latent states. The solid arrows denote stochastic connections, while the dotted arrows denote deterministic maps.

Variational Autoencoder (CVAE) [Kim et al., 2021] approach, where classification is given by $p(\mathbf{y}|\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[p_\theta(\mathbf{y}|\mathbf{z},\mathbf{x})]$. Unlike CVAE, which explicitly conditions on $\mathbf{x}$ in the decoder, VIB learns a stochastic encoder $q_\phi(\mathbf{z}|\mathbf{x})$ that ensures the latent representation $\mathbf{z}$ is a sufficient statistic for predicting $\mathbf{y}$. By enforcing this constraint, the classification model becomes more robust to noise in $\mathbf{x}$. Furthermore, unlike CVAE, VIB does not require a separate inference network $q(\mathbf{z}|\mathbf{x},\mathbf{y})$ in training.

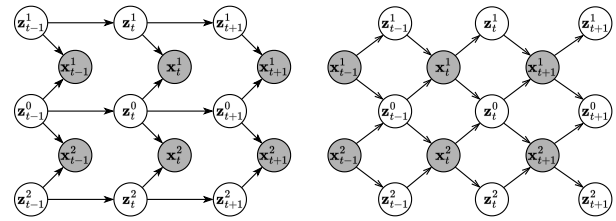
We propose DVIB as a straightforward dynamic extension to VIB and a realization of the sequence learning approach depicted in the Equation (A4) of [Fischer, 2020]. As illustrated in Figure 1, DVIB can be regarded as a stochastic counterpart of InfoNCE in the case of one-step-ahead prediction; specifically, the latent state of DVIB, $\mathbf{z}_t$, is comparable to c_t of InfoNCE, which is computed by the hidden state h_t. InfoNCE maximizes a lower bound on the mutual information between c_t and the future states known as the noise contrastive loss. By contrast, DVIB aims to learn a stochastic representation $\mathbf{z}_t$ that contains most information content of $\mathbf{x}_{1:t}$ in predicting $\mathbf{x}_{t+1}$. To achieve this, we use RNN hidden states to encode historical input $\mathbf{x}_{1:t}$, i.e. $h_t = \mathrm{RNN}(h_{t-1}, \mathbf{x}_t)$. The encoder $q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t})$ and decoder $p_\theta(\mathbf{x}_{t+1}|\mathbf{z}_t)$ networks are trained by optimizing the IB in a sequential setting,

$$\min \quad \sum_{t=1}^{T} \Big\{ \beta I(\mathbf{z}_t; \mathbf{x}_{1:t}) - I(\mathbf{z}_t; \mathbf{x}_{t+1}) \Big\}, \tag{4}$$

with a tractable variational bound:

$$\begin{aligned} \min \quad \sum_{t=1}^{T} \Big\{ & \langle \beta \log q_\phi(\mathbf{z}_t|h_t) - \log r(\mathbf{z}_t) \rangle \\ & - \langle \log p_\theta(\mathbf{x}_{t+1}|\mathbf{z}_t) \rangle \Big\}, \end{aligned} \tag{5}$$

where $r(\mathbf{z}_t) = \log \mathcal{N}(\mathbf{z}_t|0, I)$ can be chosen as the regularization term, and $\langle \cdot \rangle$ represents expectations taken with respect to $p(\mathbf{x}_{1:t+1}) q_\phi(\mathbf{z}_t|h_t)$. As will become evident later in the paper, DVIB's simplicity and the structure of its objective provide critical insights into InfoDPCCA's more general approach to modeling two sequences.



(a) DPCCA (b) InfoDPCCA

Figure 2: Graphical models for DPCCA and a simplified version of InfoDPCCA.

4 PROPOSED MODEL: INFODPCCA

In this section, we first review DPCCA [Tang et al., 2025b] and show that the proposed InfoDPCCA model is well-motivated to ameliorate a key limitation of DPCCA in capturing mutual information.

Before introducing the proposed method, we first define the notation used throughout this paper. We denote the two observations at time t as $\mathbf{x}_t^1$ and $\mathbf{x}_t^2$. The entire sequence of historical observations for variable $\mathbf{x}^i$ is represented as $\mathbf{x}_{1:t}^i$. We use $\mathbf{z}_t^0$ to denote the latent states shared by both observations and use $\mathbf{z}_t^i$ to denote latent states private to observation $\mathbf{x}^i$. Similar to the previous section, we use h_t to denote RNN hidden states, with appropriate superscripts added to distinguish between different RNNs. We use p_D to denote the empirical distribution of the dataset.

Following the notations of [Koller, 2009], we say that $\mathbf{x}$ is conditionally independent from $\mathbf{z}$ given $\mathbf{y}$, denoted as $\mathbf{x} \perp\!\!\!\perp \mathbf{z}|\mathbf{y}$, if either one of the following conditions hold:

$$p(\mathbf{x}, \mathbf{z}|\mathbf{y}) = p(\mathbf{x}|\mathbf{y}) p(\mathbf{z}|\mathbf{y}), \tag{6}$$

$$p(\mathbf{z}|\mathbf{x}, \mathbf{y}) = p(\mathbf{z}|\mathbf{y}). \tag{7}$$

4.1 DPCCA: A BASELINE FOR COMPARISON

DPCCA is a latent state-space model where a shared latent variable $\mathbf{z}_t^0$ captures mutual dependencies between the two sequences, while unique latent variables $\mathbf{z}_z^1$ and $\mathbf{z}_z^2$ account for individual variations. As is shown by the probabilistic graphical model in Figure 2 (a), DPCCA satisfies the following set of (conditional) independence properties:

$$z_{t+1}^i \perp\!\!\!\perp z_{t+1}^j | \mathbf{z}_t^{0:2}, \quad x_t^1 \perp\!\!\!\perp x_t^2 | \mathbf{z}_t^{0:2}, \quad z_t^i \not\perp\!\!\!\perp z_t^j | \mathbf{x}_t^{1:2}, \tag{8}$$

for $i, j \in \{0, 1, 2\}$. Each recurrent segment of DPCCA has the state transition and emission models

$$p_\theta(\mathbf{z}_t^{0:2}|\mathbf{z}_{t-1}^{0:2}) = \prod_{i=0}^{2} p_\theta(\mathbf{z}_t^i|\mathbf{z}_{t-1}^i), \tag{9}$$

$$p_\theta(\mathbf{x}_t^{1:2}|\mathbf{z}_t^{0:2}) = p_\theta(x_t^1|z_t^1, z_t^0) p_\theta(x_t^2|z_t^2, z_t^0), \tag{10}$$

where θ denotes the parameters of the generative process. DPCCA is trained by maximizing the variational lower bound (ELBO),

$$\max_{\theta,\phi} \quad \mathbb{E}_{q_\phi(\mathbf{z}_{1:T}^{0:2}|\mathbf{x}_{1:T}^{1:2})}\left[\log \frac{p_\theta(\mathbf{x}_{1:T}^{1:2}, \mathbf{z}_{1:T}^{0:2})}{q_\phi(\mathbf{z}_{1:T}^{0:2}|\mathbf{x}_{1:T}^{1:2})}\right], \tag{11}$$

where the inference network q_ϕ has the form

$$q_\phi(\mathbf{z}_{1:T}^{0:2}|\mathbf{x}_{1:T}^{1:2}) = \prod_{t=1}^{T} q(\mathbf{z}_t^{0:2}|\mathbf{z}_{t-1}^{0:2}, \mathbf{x}_{t:T}^{1:2}). \tag{12}$$

4.2 COMPARISON OF GRAPHICAL MODELS

For ease of comparison, we illustrate the graphical model of a simplified version of InfoDPCCA in Figure 2 (b), which assumes that the current hidden state is conditionally independent of previous observations given the current observations. However, in practice, we consider a more general case where each hidden state depends on all previous observations. The graphical model for this general case of InfoDPCCA is shown in Figure 3 (b).

It can be verified that, similar to DPCCA (8), InfoDPCCA satisfies the following conditional independence properties:

$$z_{t+1}^i \perp\!\!\!\perp z_{t+1}^j|\mathbf{x}_{t+1}^{1:2}, \quad x_t^1 \perp\!\!\!\perp x_t^2|\mathbf{z}_{t-1}^{0:2}, \quad z_t^i \not\perp\!\!\!\perp z_t^j|\mathbf{x}_{t+1}^{1:2}. \tag{13}$$

A major difference between the two models is that InfoDPCCA assumes a conditional independence property for the distribution of its latent states, which we refer to as serial independence, defined as

$$z_t^i \perp\!\!\!\perp z_{t-n}^j|\mathbf{x}_{1:T}^{1:2}, \quad \forall\, i, j \in \{0, 1, 2\}. \tag{14}$$

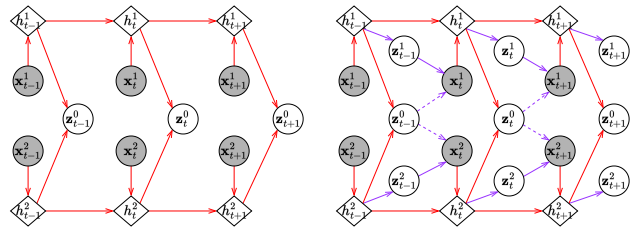
This assumption implies that given all the observations, the private latent states of different observations at different time steps are conditionally independent. The serial independence of InfoDPCCA can be established by

$$p(z_t^i, z_{t-n}^j|\mathbf{x}_{1:T}^{1:2}) = p(z_t^i|\mathbf{x}_{1:t+1}^{1:2})p(z_{t-n}^j|\mathbf{x}_{1:t-n+1}^{1:2}), \tag{15}$$

according to the criteria (6). These properties hold for both the simplified and general versions of InfoDPCCA. On the other hand, in the case of DPCCA, we can prove $z_{t-1}^1 \not\perp\!\!\!\perp z_t^2|\mathbf{x}_{1:T}^{1:2}$ using (6),

$$\begin{aligned} p(z_{t-1}^1, z_t^2|\mathbf{x}_{1:T}^{1:2}) &= \sum_{z_{t-1}^0, z_{t-1}^2} p(z_{t-1}^1, z_{t-1}^0, z_{t-1}^2, z_t^2|\mathbf{x}_{1:T}^{1:2}) \\ &= p(z_{t-1}^1|\mathbf{x}_{1:T}^{1:2}) \sum_{z_{t-1}^0, z_{t-1}^2} p(z_{t-1}^0, z_{t-1}^2, z_t^2|\mathbf{x}_{1:T}^{1:2}, z_{t-1}^1) \\ &\neq p(z_{t-1}^1|\mathbf{x}_{1:T}^{1:2})p(z_t^2|\mathbf{x}_{1:T}^{1:2}). \end{aligned}$$

Admittedly, the graphical model of DPCCA may seem more intuitive; however, we believe that the mesh structure of



(a) Step I (b) Step II

Figure 3: Graphical models in the two-step training of InfoDPCCA. The diamond nodes denote RNN hidden states. We use solid red arrows to denote the connections learned in step I, and solid purple arrows to denote the connections trained in step II, and dashed purple arrows to denote the residual connections from step I.

InfoDPCCA is necessary to facilitate information-theoretic representation learning. Generative models like VAEs or DPCCA assume that the observations are generated by the latent variables ($\mathbf{z}_t \to \mathbf{x}_t$), while representation learning methods that apply information-theoretic principles like VIB and Multi-view IB treat the latent variables as disentangled representations of the observations ($\mathbf{x}_t \to \mathbf{z}_t$). InfoDPCCA falls into the latter category, and its objective function is analyzed in detail in Section 4.3.

Furthermore, the serial independence assumption aligns with other established methods when applied to, for example, industrial processes or single-cell biomedical sequences. LaVAR-CCA [Qin, 2022] combines contemporaneous orthogonality, where latent variables are uncorrelated at each time step, with a vector autoregressive model that captures temporal dependencies. A conceptually similar approach to InfoDPCCA appears in Tilted-CCA [Lin and Zhang, 2023], which, while applied to non-sequential single-cell data, uses orthogonality constraints to separate shared and distinct information across modalities.

4.3 INFORMATION-THEORETIC OBJECTIVE

The shared latent states $\mathbf{z}_t^0$ of DPCCA have an uncertain meaning. Beyond the structural design of the generative model, there is no guarantee that the extracted shared latent states exclusively capture the mutual information between the two observations—nor is the model explicitly trained to enforce this property. For instance, if the dimensionality of $\mathbf{z}_t^0$ is relatively small, the private latent states $\mathbf{z}_t^1$ and $\mathbf{z}_t^2$ may independently encode the dynamics of $\mathbf{x}_t^1$ and $\mathbf{x}_t^2$, respectively, leading to a low reconstruction loss; in such case $\mathbf{z}_t^0$ may primarily encode residual dynamics that lack meaningful shared structure.

To remedy this, InfoDPCCA uses an information-theoretic objective to encourage the shared latent states $\mathbf{z}_t^0$ to exclusively capture the mutual information between the two

observations. We extract latent states $\mathbf{z}_t^0$ as the stochastic embeddings of $\mathbf{x}_{1:t}^{1:2}$ by optimizing the following objective,

$$\begin{aligned}\min \quad & \sum_{t=1}^{T} \Big\{ \underbrace{\alpha I(\mathbf{z}_t^0; \mathbf{x}_{1:t}^{1:2}) - I(\mathbf{z}_t^0; \mathbf{x}_{t+1}^{1:2})}_{\text{IB term}} \\ & + \beta \underbrace{\big(I(\mathbf{z}_t^0; \mathbf{x}_{1:t}^1 | \mathbf{x}_{1:t}^2) + I(\mathbf{z}_t^0; \mathbf{x}_{1:t}^2 | \mathbf{x}_{1:t}^1) \big)}_{\text{regularization term}} \Big\},\end{aligned} \tag{16}$$

where the IB term is similar to (1) and (4) and the regularization term discourage $\mathbf{z}_t^0$ to learn sequence-specific information as illustrated in Figure 4.

As depicted in Figure 3 (a), our framework assumes that $\mathbf{z}_t^0$ is generated by $\mathbf{x}_{1:t}^{1:2}$ via a stochastic map,

$$p(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2}) = q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2}) = q_0^{1:2}(\mathbf{z}_t^0 | h_t^1, h_t^2), \tag{17}$$

where $h_t^i = \mathrm{RNN}(h_t^i, \mathbf{x}_t^i)$ for $i = 1, 2$. We also assume that the emission model can be factored as

$$p(\mathbf{x}_{t+1}^{1:2} | \mathbf{z}_t^0) = p_1^0(\mathbf{x}_{t+1}^1 | \mathbf{z}_t^0) p_2^0(\mathbf{x}_{t+1}^2 | \mathbf{z}_t^0). \tag{18}$$

Following the idea of VIB, we have the following bounds

$$\begin{aligned} I(\mathbf{z}_t^0; \mathbf{x}_{1:t}^{1:2}) &\leq \langle \log q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2}) - \log r(\mathbf{z}_t^0) \rangle, \\ I(\mathbf{z}_t^0; \mathbf{x}_{t+1}^{1:2}) &\geq \langle \log p_1^0(\mathbf{x}_{t+1}^1 | \mathbf{z}_t^0) p_2^0(\mathbf{x}_{t+1}^2 | \mathbf{z}_t^0) \rangle + \text{const.}, \end{aligned} \tag{20}$$

where $\langle \cdot \rangle$ represent expectations taken with respect to $p_D(\mathbf{x}_{1:t+1}^{1:2}) q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2})$.

To further simplify the regularization term in (16), we temporarily simplify the notations by removing the superscripts of each variable and using their original superscripts as their subscripts, e.g. $\mathbf{x}_{1:t}^{1:2} \to \mathbf{x}_{1:2}$. Firstly, we have the equality

$$\begin{aligned} & I(\mathbf{z}_0; \mathbf{x}_1 | \mathbf{x}_2) + I(\mathbf{z}_0; \mathbf{x}_2 | \mathbf{x}_1) \\ &= \big(H(\mathbf{z}_0; \mathbf{x}_2) - H(\mathbf{z}_0; \mathbf{x}_{1:2}) \big) + \big(H(\mathbf{z}_0; \mathbf{x}_1) - H(\mathbf{z}_0; \mathbf{x}_{1:2}) \big) \\ &= \big(I(\mathbf{z}_0; \mathbf{x}_{1:2}) - I(\mathbf{z}_0; \mathbf{x}_2) \big) + \big(I(\mathbf{z}_0; \mathbf{x}_{1:2}) - I(\mathbf{z}_0; \mathbf{x}_1) \big). \end{aligned} \tag{21}$$

Furthermore, we can establish a variational bound

$$\begin{aligned} & I(\mathbf{z}_0; \mathbf{x}_{1:2}) - I(\mathbf{z}_0; \mathbf{x}_1) \\ &= \iint p(\mathbf{z}_0, \mathbf{x}_{1:2}) \log \frac{p(\mathbf{z}_0 | \mathbf{x}_{1:2})}{\cancel{p(\mathbf{z}_0)}} \mathrm{d}\mathbf{z}_0 \mathrm{d}\mathbf{x}_{1:2} \\ &\quad - \iint p(\mathbf{z}_0, \mathbf{x}_1) \log \frac{p(\mathbf{z}_0 | \mathbf{x}_1)}{\cancel{p(\mathbf{z}_0)}} \mathrm{d}\mathbf{z}_0 \mathrm{d}\mathbf{x}_1 \\ &\overset{(a)}{\leq} \mathbb{E}_{p(\mathbf{z}_0, \mathbf{x}_{1:2})} [\log q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2})] - \mathbb{E}_{p(\mathbf{z}_0, \mathbf{x}_1)} [\log q_0^1(\mathbf{z}_0 | \mathbf{x}_1)] \\ &\approx \langle \log q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2}) \rangle - \langle \log q_0^1(\mathbf{z}_0 | \mathbf{x}_1) \rangle, \end{aligned} \tag{22}$$

where the inequality in (a) holds due to the modeling assumption that $p(\mathbf{z}_0 | \mathbf{x}_{1:2}) = q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2})$ and the Gibbs' inequality

$$\begin{aligned} 0 &\leq \mathbb{E}_{p(\mathbf{x}_1)} [D_{\mathrm{KL}}(p(\mathbf{z}_0 | \mathbf{x}_1) \| q_0^1(\mathbf{z}_0 | \mathbf{x}_1))] \\ &= \mathbb{E}_{p(\mathbf{z}_0, \mathbf{x}_1)} [\log p(\mathbf{z}_0 | \mathbf{x}_1)] - \mathbb{E}_{p(\mathbf{z}_0, \mathbf{x}_1)} [\log q_0^1(\mathbf{z}_0 | \mathbf{x}_1)], \end{aligned} \tag{23}$$

and the expectations are defined as

$$\begin{aligned} \langle \log q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2}) \rangle &= \mathbb{E}_{p_D(\mathbf{x}_{1:t}^{1:2}) q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2})} [\log q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2})], \\ \langle \log q_0^1(\mathbf{z}_0 | \mathbf{x}_1) \rangle &= \mathbb{E}_{p_D(\mathbf{x}_{1:t}^{1:2}) q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2})} [\log q_0^1(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^1)]. \end{aligned} \tag{24}$$

Similarly, we obtain the variational bound,

$$\begin{aligned} & I(\mathbf{z}_0; \mathbf{x}_{1:2}) - I(\mathbf{z}_0; \mathbf{x}_2) \\ &\leq \langle \log q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2}) \rangle - \langle \log q_0^2(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^2) \rangle. \end{aligned} \tag{25}$$

In summary, we solve for encoder $q_0^{1:2}$, q_0^1, q_0^2, and decoder p_1^0, p_2^0 networks by optimizing the following objective,

$$\begin{aligned} \min \quad & (\alpha + 2\beta) \langle \log q_0^{1:2}(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^{1:2}) \rangle - \alpha \langle r(\mathbf{z}_t^0) \rangle \\ & - \langle \log p_1^0(\mathbf{x}_{t+1}^1 | \mathbf{z}_t^0) \rangle - \langle \log p_2^0(\mathbf{x}_{t+1}^2 | \mathbf{z}_t^0) \rangle \\ & - \beta \langle \log q_0^1(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^1) \rangle - \beta \langle \log q_0^2(\mathbf{z}_t^0 | \mathbf{x}_{1:t}^2) \rangle, \end{aligned} \tag{26}$$

where we use the bounds developed in Equations (19) and (20). The extracted shared latent states $\mathbf{z}_t^0$, as formulated above, achieve a balance between minimizing overall information content, ensuring they capture only the shared components between the two observations, and retaining sufficient information to predict the next observation.

Unfortunately, there's no closed-form solution to pick α and β. One may use cross-validation to set the hyperparameters. By default, we can set $\alpha = 1$ and $\beta = 0.1$.

Remark 1. While the Gibbs' inequality (23) holds for any choice of distribution over $\mathbf{z}_0$, the tightness of the variational bound (22) depends on how well $q_0^1(\mathbf{z}_0 | \mathbf{x}_1)$ approximates the true posterior $p(\mathbf{z}_0 | \mathbf{x}_1)$. In fact, the training objective (26) is designed to encourage this match by including the term $-\beta \langle \log q_0^1(\mathbf{z}_0 | \mathbf{x}_1) \rangle$ (as defined in (24)). Minimizing this term drives $q_0^1(\mathbf{z}_0 | \mathbf{x}_1)$ to align with $q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2})$, which serves as an approximation to $p(\mathbf{z}_0 | \mathbf{x}_1)$. To see why, note that

$$\begin{aligned} p(\mathbf{z}_0 | \mathbf{x}_1) &= \int p(\mathbf{z}_0, \mathbf{x}_2 | \mathbf{x}_1) \mathrm{d}\mathbf{x}_2 = \int p(\mathbf{x}_2 | \mathbf{x}_1) q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2}) \mathrm{d}\mathbf{x}_2 \\ &\approx \int p_D(\mathbf{x}_2 | \mathbf{x}_1) q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2}) \mathrm{d}\mathbf{x}_2 = q_0^{1:2}(\mathbf{z}_0 | \mathbf{x}_{1:2}). \end{aligned} \tag{27}$$

4.4 TWO-STEP TRAINING SCHEME

Instead of training the model in a single step, we propose a novel two-step approach for InfoDPCCA. In the first step, we extract the shared latent states that capture the mutual information between the two observations as mentioned in Section 4.3. In the second step, we train the entire generative model, including the private latent states, while keeping the parameters learned in the first step fixed.

We use variational inference to train the remaining components of the generative model, specifically the private latent states and their associated stochastic mappings. The goal is for $\mathbf{z}_t^0$ to capture the shared global dynamics of the system,

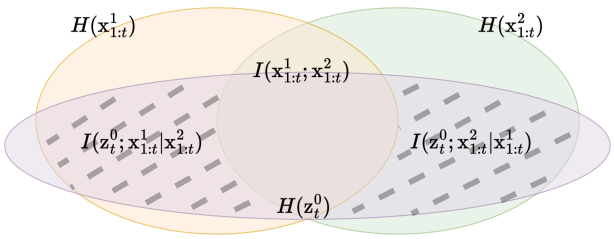


Figure 4: Venn diagram illustrating the relationship between the shared latent state $\mathbf{z}_t^0$ and the observations $\mathbf{x}_{1:t}^{1:2}$. The shaded regions represent the regularization term in (16), which InfoDPCCA aims to minimize.

while $\mathbf{z}_t^1$ and $\mathbf{z}_t^2$ encode local variations and residual fluctuations. As shown in Figure 3 (b), solid red arrows indicate the connections trained in Step I, while solid purple arrows represent the connections trained in Step II.

With the parameters of the encoder network $q_0^{1:2}$ fixed, we then solve for θ and ϕ in

$$\max \quad \mathbb{E}_{q_\phi(\mathbf{z}_{1:T}^{0:2}|\mathbf{x}_{1:T}^{1:2})}\left[\log \frac{p_\theta(\mathbf{z}_{1:T}^{0:2}, \mathbf{x}_{1:T}^{1:2})}{q_\phi(\mathbf{z}_{1:T}^{0:2}|\mathbf{x}_{1:T}^{1:2})}\right], \tag{28}$$

where

$$\begin{aligned} p_\theta(\mathbf{z}_{1:T}^{0:2}, \mathbf{x}_{1:T}^{1:2}) &= \prod_{t=1}^{T} q_0^{1:2}(\mathbf{z}_t^0|\mathbf{x}_{1:t}^{1:2}) p_\theta(\mathbf{z}_t^1|\mathbf{x}_{1:t}^1) p_\theta(\mathbf{z}_t^2|\mathbf{x}_{1:t}^2) \\ &\qquad p_\theta(\mathbf{x}_{t+1}^1|\mathbf{z}_t^{0,1}) p_\theta(\mathbf{x}_{t+1}^2|\mathbf{z}_t^{0,2}), \\ q_\phi(\mathbf{z}_{1:T}^{0:2}|\mathbf{x}_{1:T}^{1:2}) &= \prod_{t=1}^{T} q_\phi(\mathbf{z}_t^{0:2}|\mathbf{x}_{1:t+1}^{1:2}). \end{aligned} \tag{29}$$

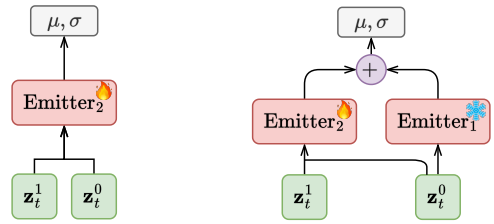
Note that we design the approximate posterior in a way that its factorization aligns in form with the true posterior factorization,

$$p(\mathbf{z}_{1:T}^{0:2}|\mathbf{x}_{1:T}^{1:2}) = \prod_{t=1}^{T} p(\mathbf{z}_t^{0:2}|\mathbf{x}_{1:t+1}^{1:2}). \tag{30}$$

In summary, the generative process is given by $p_\theta(\mathbf{z}_{1:T}, \mathbf{x}_{1:T}^{1:2})$, which can be used for sequence generation and prediction. Meanwhile, the inference model $q_\phi(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}^{1:2})$ is used to extract the latent states $\mathbf{z}_t^{0:2}$ for down-stream tasks.

4.5 IMPROVING STABILITY & EFFICIENCY

If the goal is for $\mathbf{z}_t^0$ to encode the primary or slow dynamics of the system while $\mathbf{z}_t^1$ and $\mathbf{z}_t^2$ capture the residual structures, is it optimal to reuse only the stochastic mapping $q_0^{1:2}(\mathbf{z}_t^0|\mathbf{x}_{1:t}^{1:2})$, as indicated by the purple dashed arrows in Figure 5 (b), while ignoring other inference networks trained in Step I? In this section, we explore several approaches to leveraging the learned structures from Step I. In practice, these methods enhance both stability and efficiency (reduction in the total amount of parameters) in training.



(a) New Emitter (b) Residual Connection

Figure 5: Two neural network structures for InfoDPCCA's emission model. (a) shows a new emitter network trained from scratch, with trainable parameters indicated by fire icons. (b) reuses the emitter network from Step I via a residual connection, with fixed parameters indicated by ice icons.

Similar to the other deep state space models [Girin et al., 2020, Tang et al., 2025a, Krishnan et al., 2017], we define the decoder network $p(\mathbf{x}|\mathbf{z})$ as a Gaussian distribution, where the mean and variance are parameterized by a neural network referred to as the emitter:

$$p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mu_\theta(\mathbf{z}), \sigma_\theta(\mathbf{z})^2), \tag{31}$$

where $(\mu_\theta(\mathbf{z}), \sigma_\theta(\mathbf{z})) = \text{Emitter}_\theta(\mathbf{z})$. In practice, we implement the emitter using a multi-layer perceptron (MLP). For example, we denote a three-layer MLP with ReLU activations in the hidden layers and a Sigmoid activation in the output layer as:

$$\text{Emitter}(\mathbf{z}) = \text{MLP}(\mathbf{z}, \text{ReLU}, \text{ReLU}, \text{Sigmoid}). \tag{32}$$

Residual connections. To optimize the information bottleneck objective (16) in step I, we obtain a byproduct $p_1(\mathbf{x}_{t+1}^1|\mathbf{z}_t^0)$ associated with Emitter_1. This raises the question of whether or not Emitter_1 can be reused in Step II. Inspired by ResNet [He et al., 2016], we propose two designs for the decoder network $p_\theta(\mathbf{x}_{t+1}^1|\mathbf{z}_t^0, \mathbf{z}_t^1)$, referred to as Emitter_2. Figure 5 (a) illustrates a baseline approach where a separate emitter is trained from scratch:

$$\text{Emitter}_2(\mathbf{z}_t^{0:1}) = \text{MLP}([\mathbf{z}_t^{0:1}], \text{ReLU}, \text{ReLU}, \text{Sigmoid}). \tag{33}$$

In contrast, Figure 5 (b) presents an alternative approach where a residual connection from $\text{Emitter}_1(\mathbf{z}_t^0)$ is incorporated into $\text{Emitter}_2(\mathbf{z}_t^0, \mathbf{z}_t^1)$. To adaptively control the information flow from Emitter_1, we introduce gating units [Cho et al., 2014, Krishnan et al., 2017], leading to the following parameterization:

$$\begin{aligned} [\mu_1, \sigma_1] &= \text{Emitter}_2(\mathbf{z}_t^0), \\ [\tilde{\mu}_2, \sigma_2] &= \text{MLP}([\mathbf{z}_t^0, \mathbf{z}_t^1], \text{ReLU}, \text{ReLU}, \text{Sigmoid}), \\ g_t &= \text{MLP}([\mathbf{z}_t^0, \mathbf{z}_t^1], \text{ReLU}, \text{Sigmoid}), \textit{ (gating units)} \\ \mu_2 &= (1 - g_t) \odot \mu_1 + g_t \odot \tilde{\mu}_2, \end{aligned} \tag{34}$$

Algorithm 1: Two-Step Training for `InfoDPCCA`

Input: Observed paired sequences $\{\mathbf{x}^1_{1:T}, \mathbf{x}^2_{1:T}\}$;
Hyperparameters α, β from (16).

// Step I: Optimizing info-theoretic objective;
Initialize parameters for inference networks $q_0^{1:2}$, q_0^1, q_0^2, p_1^0, p_2^0, and RNNs d^1 and d^2;
repeat
  $h^1_{1:T} \leftarrow d^1(\mathbf{x}^1_{1:T}), \quad h^2_{1:T} \leftarrow d^2(\mathbf{x}^2_{1:T})$;
  Sample $\mathbf{z}_t^0 \sim q_0^{1:2}(\mathbf{z}_t^0 | h_t^1, h_t^2)$;
  Compute the objective (26);
  Update $q_0^{1:2}, q_0^1, q_0^2, p_1^0, p_2^0, d^1, d^2$ via stochastic gradient descent algorithm;
until *convergence*;
// Step II: Full generative model training;
if `residual_connection` **then**
  Construct Emitter_2 in (34) using p_1^0 and p_2^0;
else
  Construct Emitter_2 in (33);
end
repeat
  if `reuse_RNN` **then**
    $h^1_{1:T} \leftarrow d^1(\mathbf{x}^1_{1:T}), \quad h^2_{1:T} \leftarrow d^2(\mathbf{x}^2_{1:T})$;
    Sample $\mathbf{z}_t^{0:2}$ following procedures (35);
  else
    Sample $\mathbf{z}_t^{0:2} \sim q_\phi(\mathbf{z}_t^{0:2} | \mathbf{x}^{1:2}_{1:t+1})$ in (29);
  end
  Compute the objective (28);
  Update θ, ϕ via stochastic gradient descent;
until *convergence*;
$h^1_{1:T} \leftarrow d^1(\mathbf{x}^1_{1:T}), \quad h^2_{1:T} \leftarrow d^2(\mathbf{x}^2_{1:T})$;
$\mu_t \leftarrow W_\mu[h^1_{t+1}, h^2_{t+1}] + b_\mu$;
Output: Extracted latent states $\mathbf{z}^{0:2}_{1:T} \leftarrow \mu_{1:T}$

where $\odot$ denotes element-wise product. As shown in (34), only μ_1 is directly used to compute μ_2, but we retain the flexibility to reuse both μ_1 and σ_1. The parameters of Emitter_1 remain fixed in Step II. The use of residual connections can be particularly beneficial when $\mathbf{z}_t^0$ has a strong influence on the output, allowing the contribution of $\mathbf{z}_t^1$ to be adjusted adaptively via the residual connection alone.

Inference network design. The optimization of the ELBO (28) in Step II requires a separate inference network $q_\phi(\mathbf{z}^{0:2}_{1:T} | \mathbf{x}^{1:2}_{1:T})$, which introduces redundancy in the model parameters. Inspired by Variational RNN (VRNN) [Chung et al., 2015], we can mitigate this redundancy by reusing the RNN hidden states from Step I.

Specifically, we adopt the following parameterization, which results in $q_\phi(\mathbf{z}_t^{0:2} | \mathbf{x}^{1:2}_{1:t+1}) = \mathcal{N}(\mathbf{z}_t^{0:2} | \mu_t, \text{diag}(\sigma_t^2))$ as in (29):

$$\begin{aligned} &h^1_{1:T} = d^1(\mathbf{x}^1_{1:T}), \quad h^2_{1:T} = d^2(\mathbf{x}^2_{1:T}), \\ &\mu_t = W_\mu[h^1_{t+1}, h^2_{t+1}] + b_\mu, \\ &\sigma_t = \text{Softplus}(W_\sigma[h^1_{t+1}, h^2_{t+1}] + b_\mu), \end{aligned} \tag{35}$$

where h^1_{t+1} and h^2_{t+1} are RNN hidden states, as represented by the diamond-shaped nodes in Figure 3, which together encode information from $\mathbf{x}^{1:2}_{1:t+1}$.

By applying residual connections and reusing the generative RNN in Step II, empirical evidence shows that this approach leads to improved training stability and a reduction in parameter size. A high-level algorithmic summary is provided in Algorithm 1.

5 EXPERIMENTS

In this section, we present a series of experiments to evaluate the performance of InfoDPCCA. First, we conduct numerical simulations using synthetic data generated by the Hénon map to assess the model's ability to recover latent structures. Then, we apply InfoDPCCA to real-world medical fMRI data, where we aim to demonstrate its effectiveness in extracting shared latent dynamics across different neural states and patient groups. These experiments highlight the advantages of InfoDPCCA in both synthetic and real-world applications.

5.1 NUMERICAL SIMULATION

Since there is no known creteria for validating whether a model extracts the shared latent variables $\mathbf{z}_t^0$ only encodes the mutual information between the observations $I(\mathbf{x}^1_{1:t}; \mathbf{x}^2_{1:t})$, we designed a synthetic dataset with known latent states and developed a metric that measures how similar the extracted shared latent states are to the ground truth.

The dataset is generated using the Hénon map, a two-dimensional chaotic system defined by $x_{t+1} = 1 - 1.4x_t^2 + y_t$ and $y_{t+1} = 0.3x_t$. Each sequence starts from a random initial condition (x_0, y_0) and evolves for T time steps. The latent states are linearly mapped to observed variables $\mathbf{x}_t \in \mathbb{R}^{d_x}$ and $\mathbf{y}_t \in \mathbb{R}^{d_y}$ using randomly initialized matrices, with added Gaussian noise. The dataset consists of N sequences and is split into 80% training and 20% testing.

We can calculate the Pearson correlation coefficient between the true latent states $\mathbf{z}$ and the extracted ones $\hat{\mathbf{z}}$. Let $\mathbf{z}_t, \hat{\mathbf{z}}_t \in \mathbb{R}^D$. We define global-mean correlation metric as

$$\rho_{jk} := \frac{\sum_{n=1}^N \sum_{t=1}^T (z_{ntj} - \overline{z_j})(\hat{z}_{ntk} - \overline{\hat{z}_k})}{\sqrt{\sum_{n=1}^N \sum_{t=1}^T (z_{ntj} - \overline{z_j})^2} \sqrt{\sum_{n=1}^N \sum_{t=1}^T (\hat{z}_{ntk} - \overline{\hat{z}_k})^2}},$$

where

$$\overline{z_j} = \frac{1}{NT} \sum_{n=1}^N \sum_{t=1}^T z_{ntj}, \quad \overline{\hat{z}_k} = \frac{1}{NT} \sum_{n=1}^N \sum_{t=1}^T \hat{z}_{ntk}.$$

Finally, we report the sum of maximum correlations

$$\hat{\rho} = \frac{1}{D} \sum_{j=1}^D \max_{k \in [D]} |\rho_{jk}|. \tag{36}$$

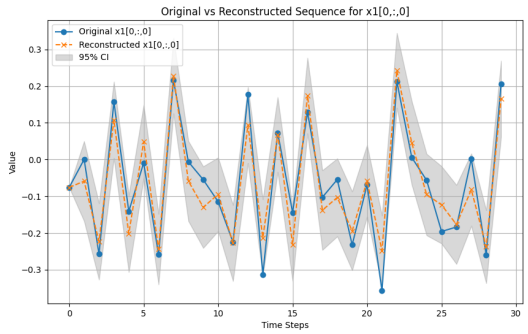


Figure 6: Reconstruction on an Hénon map using InfoDPCCA.

Note that metrics like RMSE would not work here because, for instance, a model that extracts the latent states exactly like the ground truth but with an opposite sign would fail the RMSE test.

A visual inspection on the reconstruction can be seen in Figure 6. As can be seen, the reconstructed mean closely monitors the actual data, and every data point is confined within the predicted confidence interval.

The step I of InfoDPCCA is an info-theoretic representation learning method, while step II is a standard generative model that creates a separate inference network and maximizes an ELBO objective. The global-mean correlation metric for InfoDPCCA with only step II is 65% and the full two-step InfoDPCCA achieves a score of 72%, which demonstrates the effectiveness of step I in capturing the underlying dynamics.

5.2 MEDICAL DATA

We evaluate our approach using three publicly available datasets, which consist of resting - state functional magnetic resonance imaging (rs-fMRI) data corresponding to different neural states or mental disorders. Our hypothesis is that the rs-fMRI data and its corresponding blood oxygen level - dependent (BOLD) signals can be modelled as a nonlinear dynamic system. Moreover, we assume that the common dynamics or the main driving forces corresponding to different neural states are distinguishable, and these could be further exploited as potential biomarkers for future neuroscience research.

The first dataset is the eyes closed and eyes open (ECEO) dataset [Stefan, 2020]. It contains rs-fMRI data of 48 college students (22 females) aged between 19 and 31 years. These students were scanned in both the eyes open and eyes closed states, yielding a total of 96 samples. The task is to differentiate between the two states using the rs-fMRI data. The second dataset is from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database [Initiative, 2025]. Here, the objective is to distinguish the Alzheimer's Disease (AD) group, consisting of 37 patients, from the normal control (NC) subjects, with a total of 37 individuals. The third dataset is from the Autism Brain Imaging Data Exchange I (ABIDE - I) [Di Martino et al., 2014]. This dataset contains rs-fMRI data of 184 patients with Autism Spectrum Disorder (ASD) and 571 typically developed (TD) individuals, collected from 17 sites. In this paper, we utilize data collected from NYU and choose 40 ASD and 40 TD samples. For each dataset, the Automated Anatomical Labeling (AAL) template was employed to extract the region-of-interest (ROI)-averaged time series from 116 ROIs. In other words, the rs-fMRI data was transformed into a 116-dimensional time series.

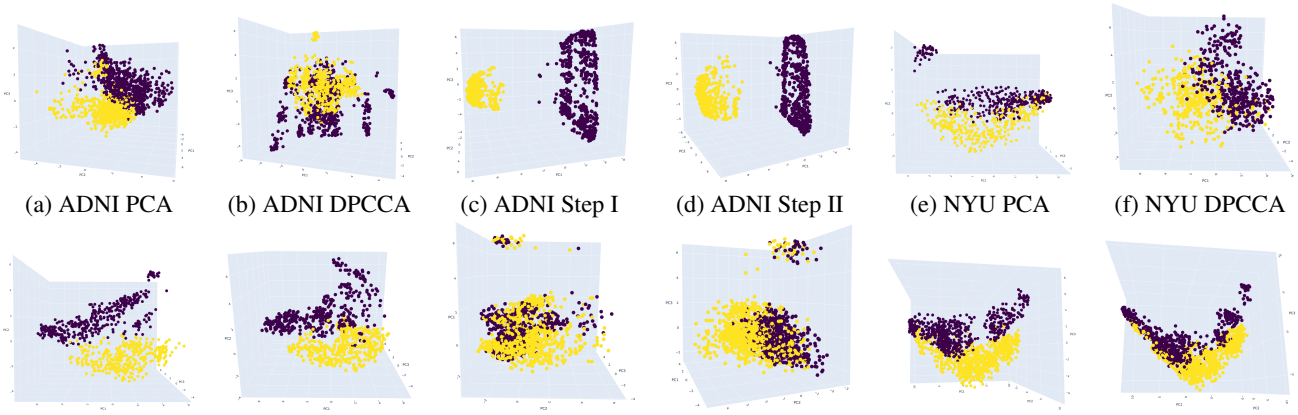
We utilize our proposed InfoDPCCA to extract shared latent states from pairs of rs-fMRI data within the same group (for example, both are from the eye-closed state or both are from schizophrenia patients). Since there is not yet a criterion to verify the claim that the extracted $\mathbf{z}^0_{1:T}$ indeed captures only the mutual information, we rely on the clustering task as a proxy.

After the common dynamics are extracted, we utilize Time Series Cluster Kernel (TCK) [Mikalsen et al., 2018] to project the multivariate times series onto a lower dimensional space where clustering tasks can be administered. Specifically, TCK clusters multivariate time series (MTS) by leveraging an ensemble of Gaussian Mixture Models (GMMs) to generate a positive semi-definite kernel matrix, capturing temporal dependencies and enabling kernel-based clustering methods. If a model is effective enough in extracting the common dynamics, we anticipate a distinct classification boundary.

Apart from DPCCA and InfoDPCCA, we add two more models for comparison:

1. PCA: Since directly classifying the raw data is not an option due to the large dimensions of time series data, we first perform dimension reductions using PCA before calculating the kernel matrix with TCK.
2. Step I of InfoDPCCA: The shared latent states sampled from step I can be viewed as their prior distribution $p(\mathbf{z}^0_t|\mathbf{x}^{1:2}_{1:t})$, while $\mathbf{z}^0_t$ sampled after step II can be regarded as posterior distribution $p(\mathbf{z}^0_t|\mathbf{x}^{1:2}_{1:T})$.

We visualize the extracted latent representations using kernel principal component analysis (KPCA) [Schölkopf et al., 1997] and compare the clustering results of PCA, DPCCA, Step I and II of InfoDPCCA across different datasets. Figure 7 presents the 3D scatter plots of the learned representations, where each point corresponds to a subject's extracted latent dynamics, and colors represent different neural states or patient groups. As shown in the figure, InfoDPCCA models yield a more structured and separable representation space compared to PCA and DPCCA. This indicates that



(a) ADNI PCA (b) ADNI DPCCA (c) ADNI Step I (d) ADNI Step II (e) NYU PCA (f) NYU DPCCA

(g) NYU Step I (h) NYU Step II (i) ECEO PCA (j) ECEO DPCCA (k) ECEO Step I (l) ECEO Step II

Figure 7: Visual inspection of clustering tasks on three datasets (ADNI, NYU, ECEO) using four models (PCA, DPCCA, Step I, and Step II of InfoDPCCA). Subfigures (a)–(d) show results for ADNI, (e)–(h) for NYU, and (i)–(l) for ECEO.

	ADNI	NYU	ECEO
PCA	0.592/0.639	0.588/0.662	0.103/0.514
DPCCA	0.301/0.587	0.417/0.600	0.197/0.240
Step I	**1.000**/0.907	**0.865/0.726**	**0.572/0.662**
Step II	**1.000/0.908**	0.642/0.620	0.228/0.586

Table 1: Numeric evaluation of the clustering task on three datasets (ADNI, NYU, ECEO) using four models (PCA, DPCCA, Step I, and Step II of InfoDPCCA). The bolded number indicates the best performer in its respective column. Each cell contains two metrics, NMI/Silhouette score (higher values indicate better performance).

InfoDPCCA better captures the underlying shared dynamics within each group, leading to improved class separability. Furthermore, the scatterplots generated by Step I and II of InfoDPCCA are similar, which aligns with the claim that they correspond to the prior and posterior distributions of the latent states.

To perform a quantitative assessment, we apply TCK to the extracted latent states and evaluate the clustering performance using Normalized Mutual Information (NMI) as external metric and Silhouette score as internal metric. NMI measures the agreement between predicted clusters and true labels, while the Silhouette score evaluates cluster cohesion and separation. Specifically, we train a classifier on 30% labeled data and compute clustering metrics on its predictions for the remaining 70% data. Table 1 summarizes the results, demonstrating that InfoDPCCA models consistently outperform PCA and DPCCA in all three datasets.

6 SUMMARY

We introduced InfoDPCCA, an information-theoretic extension of DPCCA that extracts shared and private latent representations from sequential data. By enforcing an information bottleneck objective and a two-step training scheme, InfoDPCCA improves interpretability and robustness. Experimental results on synthetic and fMRI data demonstrate its effectiveness in capturing meaningful latent structures.

We list a few future directions:

1. Hyperparameter tuning: Selecting optimal values for α and β.
2. Evaluation criteria: Developing better metrics beyond correlation coefficients to validate claims.
3. Multiset extension: Extending InfoDPCCA to multiple data streams, similar to DPCCA.
4. Right now, InfoDPCCA can be regarded as an unsupervised approach, which means they are unaware of the labels to each sample in the training process. As opposed to the approach described in [Kusano et al., 2019], which trains a conditional generative model, InfoDPCCA is not motivated to separate different groups in training. We leave the supervised extension of InfoDPCCA as well as its applications to identifying biomarkers in neuroscience as future work.

These directions will enhance the model's flexibility and applicability.

Acknowledgements

This work was funded in part by the Research Council of Norway under grant 309439. The authors would like to express their sincere gratitude to the anonymous reviewers, Dr. Hongzong Li (CityU HK), and Dr. Haoxuan Li (Peking University) for their thorough and constructive feedback, which significantly improved the quality of this work.

References

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

Nadav Amir, Stas Tiomkin, and Naftali Tishby. Past-future information bottleneck for linear feedback systems. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 5737–5742. IEEE, 2015.

Francis R Bach and Michael I Jordan. A probabilistic interpretation of canonical correlation analysis. 2005.

Gal Chechik, Amir Globerson, Naftali Tishby, and Yair Weiss. Information bottleneck for gaussian variables. *J. Mach. Learn. Res.*, 6:165–188, December 2005. ISSN 1532-4435.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics, 2014. doi: 10.3115/v1/D14-1179.

Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. *Advances in neural information processing systems*, 28, 2015.

Felix Creutzig, Amir Globerson, and Naftali Tishby. Past-future information bottleneck in dynamical systems. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 79(4):041925, 2009.

A. Di Martino, A. Scheres, D. S. Margulies, and et al. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular Psychiatry*, 19(6):659–667, 2014. doi: 10.1038/mp.2014.54. URL https://www.nature.com/articles/mp.2014.54.

Marco Federici, Anjan Dutta, Patrick Forré, Nate Kushman, and Zeynep Akata. Learning robust representations via multi-view information bottleneck. *arXiv preprint arXiv:2002.07017*, 2020.

Marco Federici, Patrick Forré, Ryota Tomioka, and Bastiaan S Veeling. Latent representation and simulation of markov processes via time-lagged information bottleneck. *arXiv preprint arXiv:2309.07200*, 2023.

Ian Fischer. The conditional entropy bottleneck. *Entropy*, 22(9):999, 2020.

Laurent Girin, Simon Leglaive, Xiaoyu Bie, Julien Diard, Thomas Hueber, and Xavier Alameda-Pineda. Dynamical variational autoencoders: A comprehensive review. *arXiv preprint arXiv:2008.12595*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 630–645. Springer, 2016.

Alzheimer's Disease Neuroimaging Initiative. The alzheimer's disease neuroimaging initiative, 2025. URL https://adni.loni.usc.edu/. Accessed: 2025-02-11.

Damjan Kalajdzievski, Ximeng Mao, Pascal Fortier-Poisson, Guillaume Lajoie, and Blake Richards. Transfer entropy bottleneck: Learning sequence to sequence information transfer. *arXiv preprint arXiv:2211.16607*, 2022.

Jaehyeon Kim, Jungil Kong, and Juhee Son. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*, pages 5530–5540. PMLR, 2021.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Arto Klami and Samuel Kaski. Probabilistic approach to detecting dependencies between data sets. *Neurocomputing*, 72(1-3):39–46, 2008.

Artemy Kolchinsky, Brendan D Tracey, and David H Wolpert. Nonlinear information bottleneck. *Entropy*, 21(12):1181, 2019.

Daphane Koller. Probabilistic graphical models: Principles and techniques, 2009.

Rahul Krishnan, Uri Shalit, and David Sontag. Structured inference networks for nonlinear state space models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Koki Kusano, Tetsuo Tashiro, Takashi Matsubara, and Kuniaki Uehara. Deep generative state-space modeling of fmri images for psychiatric disorder diagnosis. In *2019 international joint conference on neural networks (IJCNN)*, pages 1–7. IEEE, 2019.

Kevin Z Lin and Nancy R Zhang. Quantifying common and distinct information in single-cell multimodal data with tilted canonical correlation analysis. *Proceedings of the National Academy of Sciences*, 120(32):e2303647120, 2023.

Rui Meng, Tianyi Luo, and Kristofer Bouchard. Compressed predictive information coding. *arXiv preprint arXiv:2203.02051*, 2022.

Karl Øyvind Mikalsen, Filippo Maria Bianchi, Cristina Soguero-Ruiz, and Robert Jenssen. Time series cluster kernel for learning similarities between multivariate time series with missing data. *Pattern Recognition*, 76: 569–581, 2018.

Kieran A Murphy, Zhuowen Yin, and Dani S Bassett. Which bits went where? past and future transfer entropy decomposition with the information bottleneck. *arXiv preprint arXiv:2411.04992*, 2024.

Mihalis A Nicolaou, Vladimir Pavlovic, and Maja Pantic. Dynamic probabilistic cca for analysis of affective behavior and fusion of continuous annotations. *IEEE transactions on pattern analysis and machine intelligence*, 36(7): 1299–1311, 2014.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Amichai Painsky, Meir Feder, and Naftali Tishby. Nonlinear canonical correlation analysis: A compressed representation approach. *arXiv preprint arXiv:2502.05155*, 2025.

S Joe Qin. Latent vector autoregressive modeling and feature analysis of high dimensional and noisy data from dynamic systems. *AIChE Journal*, 68(6):e17703, 2022.

B. Schölkopf, A. Smola, and K.-R. Müller. Kernel principal component analysis. In *Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems (NIPS 1997)*, pages 582–588, 1997. URL https://papers.nips.cc/paper/1997/hash/5b4d9034971fd4f13f315bdb2f8ad56d-Abstract.html.

Z. Stefan. A toolbox for brain network construction and classification, 2020. URL https://github.com/zzstefan/BrainNetClass. Accessed: 2025-02-11.

Heung-Il Suk, Chong-Yaw Wee, Seong-Whan Lee, and Dinggang Shen. State-space model with deep learning for functional dynamics estimation in resting-state fmri. *NeuroImage*, 129:292–307, 2016.

Shiqin Tang, Pengxing Feng, Shujian Yu, Yining Dong, and S Joe Qin. A hierarchical taxonomy for deep state space models. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025a.

Shiqin Tang, Shujian Yu, Yining Dong, and S. Joe Qin. Deep dynamic probabilistic canonical correlation analysis. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2025b. doi: 10.1109/ICASSP49660.2025.10887585.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Weiran Wang, Xinchen Yan, Honglak Lee, and Karen Livescu. Deep variational canonical correlation analysis. *arXiv preprint arXiv:1610.03454*, 2016.

InfoDPCCA: Information-Theoretic Dynamic Probabilistic Canonical Correlation Analysis (Supplementary Material)

Shiqin Tang[1,2] **Shujian Yu**[3,4]

[1]Dept. of Data Science , City University of Hong Kong, Hong Kong
[2]Inst. of Artificial Intelligence , Peking University, Beijing, China
[3]Dept. of Computer Science , Vrije Universiteit Amsterdam, Amsterdam, Netherlands
[4]Dept. of Physics and Technology , UiT The Arctic University of Norway, Tromsø, Norway

A RELATED METHOD

In this section, we discuss a method related to InfoDPCCA named Multi-View Information Bottleneck (MVIB) [Federici et al., 2020]. Multi-view IB is an extension of the classical Information Bottleneck framework tailored to settings with multiple perspectives of the same underlying entity. Let $\mathbf{v}_1$ and $\mathbf{v}_2$ denote two distinct views of a common object. The primary goal is to derive stochastic mappings $q_\theta(\mathbf{z}_1|\mathbf{v}_1)$ and $q_\psi(\mathbf{z}_2|\mathbf{v}_2)$, such that the resulting latent representations $\mathbf{z}_1$ and $\mathbf{z}_2$ encapsulate solely the mutual information $I(\mathbf{v}_1, \mathbf{v}_2)$ shared between the two views. Multi-View IB operates as an unsupervised representation learning technique, with the learned representations being applicable to downstream tasks such as classification.

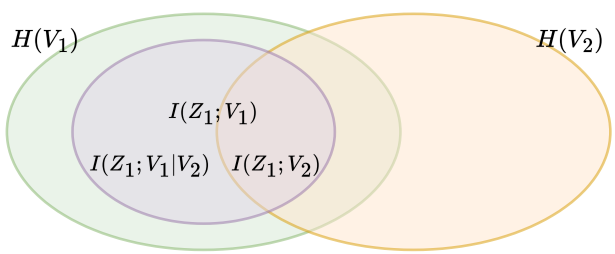


Figure 8: Venn diagram illustrating the relationship between the feature of the view $\mathbf{z}_1$ and the two views $\mathbf{v}_1$ and $\mathbf{v}_2$.

Similar to the settings of VIB, Multi-view IB assumes a factorized joint distribution:

$$p(\mathbf{v}_1, \mathbf{v}_2, \mathbf{z}_1, \mathbf{z}_2) = p(\mathbf{v}_1, \mathbf{v}_2) q_\theta(\mathbf{z}_1|\mathbf{v}_1) q_\psi(\mathbf{z}_2|\mathbf{v}_2).$$

For each view, the optimization objective is defined as:

$$\min \quad I(\mathbf{z}_1; \mathbf{v}_1|\mathbf{v}_2) - \lambda I(\mathbf{z}_1; \mathbf{v}_2),$$

where $I(\cdot;\cdot|\cdot)$ represents conditional mutual information, and λ is a trade-off parameter balancing the two terms; an illustration of this objective is shown in Fig. 8. In contrast to the supervised nature of VIB, which emphasizes label prediction, Multi-View IB seeks to maximize $I(\mathbf{z}_1; \mathbf{v}_2)$. This term captures the shared structure between views, potentially including information not directly relevant to a specific supervised task.

We summarize the similarities and differences between the MVIB and InfoDPCCA, with respective graphical models shown in Fig. 4 and Fig. 8:

- The features learned from MVIB, $\mathbf{z}_1$ and $\mathbf{z}_2$, are representations of observations $\mathbf{v}_1$ and $\mathbf{v}_2$ respectively; while in InfoDPCCA, the common latent factor $\mathbf{z}^0$ is a representation of both observations $\mathbf{x}^{1:2}$.

Accepted for the 41st Conference on Uncertainty in Artificial Intelligence (UAI 2025).

- The two methods have the same goal of learning representations that capture the mutual information between the observations while discarding the redundant information.
- InfoDPCCA uses mainly variational approximations, while MVIB relies on auxiliary mutual information estimators.

B EXPERIMENTAL SETUP ON NUMERICAL SIMULATION

The synthetic dataset is generated using the Hénon map, a chaotic dynamical system, to model a 2-dimensional ground truth latent $\mathbf{z}_t \in \mathbb{R}^2$. For each of 1000 sequences, the latent state evolves over $T = 300$ time steps via the Hénon map:

$$\mathbf{z}_{t+1} = [1 - 1.4z_{t,0}^2 + z_{t,1}, 0.3z_{t,0}]^\top,$$

with initial condition $z_{0,0} \in \text{Unif}(-1, 1)$ and $z_{0,1} \in \text{Unif}(-0.1, 0.1)$. High-dimensional observations $\mathbf{x}_t, \mathbf{y}_t \in \mathbb{R}^{120}$ are then created by projecting $\mathbf{z}_t$ through random linear transformation $W_x, W_y \in \mathbb{R}^{120\times 2}$ (drawn from $\mathcal{N}(0, 1)$), followed by additive Gaussian noise:

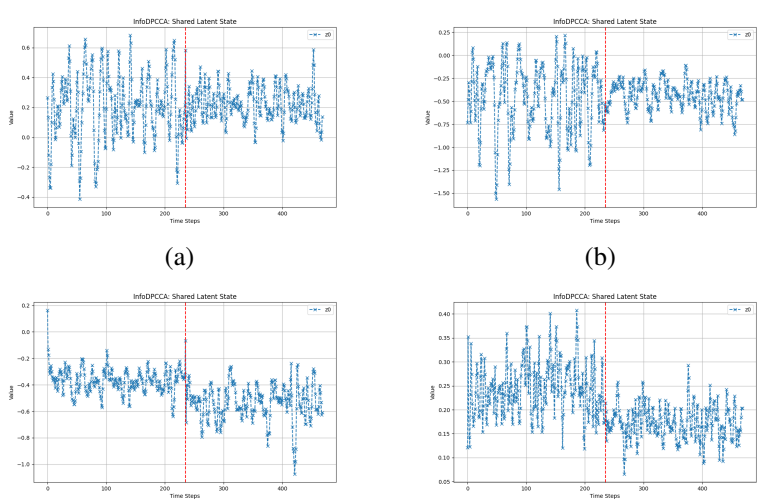
$$\mathbf{x}_t = W_x\mathbf{z}_t + \epsilon_x, \quad \mathbf{y}_t = W_y\mathbf{z}_t + \epsilon_y,$$

where $\epsilon_x, \epsilon_y \in \mathcal{N}(0, 0.05^2 I_{120})$. The dataset, comprising 1000 sequences of paired $(\mathbf{x}_t, \mathbf{y}_t)$ with corresponding $\mathbf{z}_t$, is split into 80% training and 20% testing subsets. The global-mean correlation metric in Equation (36) measures a model's ability to capture shared latent dynamics by averaging correlation coefficient ρ_{jk} between true latent states z_{ntj} (j-th dimension of $\mathbf{z}_n$ at t-th time step) and estimated states $\hat{z}_{ntk}$ across all sequences $n \in [N]$ and time steps $t \in [T]$.

C ADDITIONAL EXPERIMENTS ON FMRI DATASET

As mentioned in the paper, the ECEO dataset contains fMRI data of 48 subjects recorded with their eyes open and closed. We extract the shared latent states between EC and EO of the same subject, and we hypothesize that if the extracted shared latent states are identity-specific, they should be able to detect the change in identity. To make our setting more clear, we use math notations. Suppose we have dataset $\mathcal{D} = \{\text{EC}_i, \text{EO}_i : i \in [48]\}$. We denote $x_i^1 = \text{Concat}(\text{EC}_i, \text{EC}_{2i+1})$ and $x_i^2 = \text{Concat}(\text{EO}_i, \text{EO}_{2i+1})$. We then transform the dataset to $\mathcal{D}' = \{x_i^1, x_i^2 : i \in [24]\}$ and use it to train InfoDPCCA. We found that the shared latent states of InfoDPCCA are able to detect the change of identity by:

- Altering the magnitudes: Figure 9 (a) and (b).
- Shifting the means: Figure 9 (c) and (d).



(a) (b) (c) (d)

Figure 9: Extracted shared latent states for the supplementary experiment on the ECEO dataset.