COMPRESSED PREDICTIVE INFORMATION CODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Unsupervised learning plays an important role in many fields, such as machine learning, data compression, and neuroscience. Compared to static data, methods for extracting low-dimensional structure for dynamic data are lagging. We developed a novel information-theoretic framework, Compressed Predictive Information Coding (CPIC), to extract predictive latent representations from dynamic data. Predictive information quantifies the ability to predict the future of a time series from its past. CPIC selectively projects the past (input) into a low dimensional space that is predictive about the compressed data projected from the future (output). The key insight of our framework is to learn representations by balancing the minimization of compression complexity with maximization of the predictive information in the latent space. We derive tractable variational bounds of the CPIC loss by leveraging bounds on mutual information. The CPIC loss induces the latent space to capture information that is maximally predictive of the future of the data from the past. We demonstrate that introducing stochasticity in the encoder and maximizing the predictive information in latent space contributes to learning more robust latent representations. Furthermore, our variational approaches perform better in mutual information estimation compared with estimates under the Gaussian assumption commonly used. We show numerically in synthetic data that CPIC can recover dynamical systems embedded in noisy observation data with low signal-to-noise ratio. Finally, we demonstrate that CPIC extracts features more predictive of forecasting exogenous variables as well as auto-forecasting in various real datasets compared with other state-of-the-art representation learning models. Together, these results indicate that CPIC will be broadly useful for extracting low-dimensional dynamic structure from high-dimensional, noisy time-series data.

1 INTRODUCTION

Unsupervised methods play an important role in learning representations that provide insight into data and exploit unlabeled data to improve performance in downstream tasks in diverse application areas Bengio et al. (2013); Chen et al. (2020); Grill et al. (2020); Devlin et al. (2018); Brown et al. (2020); Baevski et al. (2020); Wang et al. (2020). Prior work on unsupervised representation learning can be broadly categorized into generative models such as variational autoencoders(VAEs) Kingma & Welling (2013) and generative adversarial networks (GAN) Goodfellow et al. (2014), discriminative models such as dynamical components analysis (DCA) Clark et al. (2019), contrastive predictive coding (CPC) Oord et al. (2018), and deep autoencoding predictive components (DAPC) Bai et al. (2020). Generative models focus on capturing the joint distribution between representations and inputs, but are usually computationally expensive. On the other hand, discriminative models emphasize capturing the dependence of data structure in the low-dimensional latent space, and are therefore easier to scale to large datasets.

In the case of sequential (i.e., dynamic) data, some representation learning models take advantage of an estimate of mutual information between encoded past (input) and the future (output) Creutzig & Sprekeler (2008); Creutzig et al. (2009); Oord et al. (2018). Although previous models utilizing mutual information extract low-dimensional representations, they tend to be sensitive to noise in the observational space. DCA directly makes use of the mutual information between the past and the future (i.e., the predictive information Bialek et al. (2001)) in a latent representational space that is a linear embedding of the observation data. However, DCA operates under Gaussian assumptions

for mutual information estimation. Later, DAPC Bai et al. (2020) extends DCA to make use of nonlinear representations by leveraging neural networks. Other popular approaches in the unsupervised representation learning literature are to use deep encoder networks to model nonlinear relations between latent representations and observed data Chen et al. (2020); Bai et al. (2020); He et al. (2020). However, use of nonlinear encoders induced by neural networks hinders computational efficiency Wang et al. (2019). Our work focus on the efficient representation learning framework for sequential data that encodes data with maximal predictive information.

We formalize our problem in terms of data generated from a stationary dynamical system and propose an information-theoretic objective function for Compressed Preditive Information Coding (CPIC). Instead of leveraging the information bottleneck (IB) objective directly as in Creutzig & Sprekeler (2008) and Creutzig et al. (2009), where the past latent representation is directly used to predict future observations, we predict the compressed future observations filtered by the encoder. It is because that in the time series setting, future observations are noisy, and treating them as labels is not insightful. Specifically, our target is to extract latent representation which can better predict future underlying dynamics. Since the compressed future observations are assumed to only retain the underlying dynamics, better compression thus contributes to extracting better dynamical representation. In addition, inspired by Clark et al. (2019) and Bai et al. (2020), we extend the prediction from single input to a window of inputs to handle high order predictive information.

Moreover, instead of directly estimating the objective information with Gaussian assumption Creutzig & Sprekeler (2008); Creutzig et al. (2009); Clark et al. (2019); Bai et al. (2020), we developed variational bounds and a tractable end-to-end training framework based on the neural estimator of mutual information studied in Poole et al. (2019). Note that our inference first leverages the variational boundary technique for self-supervised learning on the time series data. Since it alleviates the Gaussian assumption, it is applicable to a much larger class of dynamical systems.

In CPIC, we also demonstrate that introducing stochasticity into either a linear or nonlinear encoder robustly contributes to numerically better representations in different tasks. In particular, we illustrate that CPIC can recover trajectories of a chaotic dynamical system embedded in high-dimensional noisy observations with low signal-to-noise ratios in synthetic data. Furthermore, we conduct numerical experiments on four real-world datasets with different goals. In two neuroscience datasets, monkey motor cortex (M1) and rat dorsal hippocampus (HC), compared with the state-of-the-art methods, we show that the latent representations extracted from CPIC have better forecasting accuracy for the exogenous variables of the monkey's future hand position for M1, and for the rat's future position for HC. In two other real datasets, historical hourly weather temperature data (TEMP) and motion sensor data (MS), we show that latent representations extracted by CPIC have better forecasting accuracy of the future of those time series than other methods. In summary, the primary contributions of our paper are as follows:

- We developed a novel information-theoretic self-supervised learning framework, Compressed Predictive Information Coding (CPIC), which extracts low-dimensional latent representation from sequential (i.e., time series) data. CPIC maximizes the predictive information in the latent space while minimizing the compression complexity.

- We introduced the stochastic encoder structure where we encode inputs into stochastic representations to handle uncertainty and contribute to better representations.

- We derived the variational bounds of the CPIC's objective function and a tractable, end-to-end training procedure. Since our inference alleviates the Gaussian assumption common to other methods, it is applicable to a much larger class of dynamical systems. Moreover, to the best of our knowledge, our inference is the first to leverage the variational boundary technique for self-supervised learning on time series data.

- We demonstrated that, compared with the other unsupervised based methods, CPIC more robustly recovers latent dynamics in dynamical system with low signal-to-noise ratio in synthetic experiments, and extracts more predictive features for downstream tasks in various real datasets.

2 RELATED WORK

Mutual information (MI) plays an important role in estimating the relationship between pairs of variables. It is a reparameterization-invariant measure of dependency:

$$I(X,Y) = \mathbb{E}_{p(x,y)}\left[\log \frac{p(x|y)}{p(x)}\right] \tag{1}$$

It is widely used in computational neuroscience Dimitrov et al. (2011), visual representation learning Chen et al. (2020), natural language processing Oord et al. (2018) and bioinformatics Lachmann et al. (2016). In representation learning, the mutual information between inputs and representations is used to quantify the quality of the representation and is also closely related to reconstruction error in the generative models Kingma & Welling (2013); Makhzani et al. (2015). Estimating the mutual information is computationally and statistically challenging except in two cases: discrete data, as in Tishby et al. (2000) and Gaussian data, as in Chechik et al. (2005). However, these assumptions both severely constrain the class of learnable models Alemi et al. (2016). Recent works leverage deep learning models to obtain both differentiable and scalable MI estimation Belghazi et al. (2018); Nguyen et al. (2010); Oord et al. (2018); Alemi et al. (2016); Poole et al. (2019); Cheng et al. (2020).

In terms of representation learning in sequential data, Creutzig & Sprekeler (2008) utilized the information bottleneck (IB) method Tishby et al. (2000) and developed an information-theoretic objective function. Creutzig et al. (2009) proposed an alternative objective function based on a specific state-space model. Recently, Oord et al. (2018) proposed the CPC to extract dynamic information based on an autoregressive model on representations and contrastive loss on predictions. Clark et al. (2019) proposed DCA, an unsupervised learning approach to extract a subspace with maximal dynamical structure. DCA assumes that data are stationary multivariate Gaussian process and consider a linear mapping to compress data into a low-dimensional subspace and then maximize the predictive information (PI). Due to the Gaussian assumption, the mutual information can be expressed in a differentiable closed form with respect to model parameters Chechik et al. (2005). All of the above unsupervised representation learning models, except for CPC, assume the data to be Gaussian, which may be not realistic, especially when applied to neuroscience datasets O'Doherty et al. (2017); Glaser et al. (2020), given the non-Gaussianity of neuronal activity. Here, we leverage recently introduced neural estimation of mutual information to construct lower bounds of the CPIC objective and develop an end-to-end training procedure. CPIC enables generalization beyond the Gaussian case and beyond autoregressive models.

3 COMPRESSED PREDICTIVE INFORMATION CODING

The main intuition behind Compressed Predictive Information Coding (CPIC) is to extract low dimensional representations with minimal compression complexity and maximal dynamical structure. Specifically, CPIC first discards low-level information and noise that is more local by minimizing compression complexity between inputs and representations to improve model generalization. Second, CPIC maximizes the predictive information in the latent space of compressed representations. In the literature, there are different approaches to quantify the predictive capability: Wiskott & Sejnowski (2002); Turner & Sahani (2007) targeted the slowly varying features, while Creutzig & Sprekeler (2008); Creutzig et al. (2009) maximized the predictive information between past input and future input at a single time step. Recently, Clark et al. (2019); Bai et al. (2020) extended the prediction for a single point to the prediction for a range of temporal window sizes of points in the estimation of predictive information. However, the predictive information based methods above are estimated based on the Gaussian assumption. Compared with Clark et al. (2019); Bai et al. (2020), CPIC first utilizes stochastic encoder to handle uncertainty of representations, which contributes to more robust representations, and also relieves the Gaussian assumption by constructing bounds of mutual information based on neural estimations.

In more detail, instead of employing a deterministic linear mapping function as the encoder to compress data as in Clark et al. (2019), CPIC takes advantage of a stochastic linear or nonlinear mapping function. Given inputs, the stochastic representation follows Gaussian distributions, with means and variances encoded from any neural network structure. A nonlinear CPIC refers to a stochastic nonlinear encoder including a nonlinear mean encoder and a linear variance encoder, while a linear

CPIC refers to a stochastic linear encoder in which it replaces the linear mean encoder by a nonlinear mean encoder. Note that stochastic representations conditioned on inputs are parameterized as a conditional Gaussian distribution, but the marginal distribution of the representation is a mixture of Gaussian distribution, which is widely recognized as universal approximator of densities. On the other hand, avoiding the Gaussian assumption on mutual information Creutzig & Sprekeler (2008); Creutzig et al. (2009); Clark et al. (2019); Bai et al. (2020), CPIC leverages neural estimations of mutual information. Specifically, we propose differentiable and scalable bounds of the CPIC objective via variational inference, which enables end-to-end training.

Formally, let $X = \{x_t\}, x_t \in \mathbb{R}^N$ be a stationary, discrete time series, and let $X_{\text{past}} = (x_{-T+1}, \ldots, x_0)$ and $X_{\text{future}} = (x_1, \ldots, x_T)$ denote consecutive past and future windows of length T. Then both past and future data are compressed into past and future representations denoted as $Y_{\text{past}} = (y_{-T+1}, \ldots, y_0)$ and $Y_{\text{future}} = (y_1, \ldots, y_T)$ with embedding dimension size Q. Similar to the information bottleneck (IB) Tishby et al. (2000), the CPIC objective contains a trade-off between two factors. The first seeks to minimize the compression complexity and the second to maximize the predictive information in the latent (representation) space. Note that when the encoder is deterministic the compression complexity is deprecated and when the encoder is stochastic the complexity is measured by the mutual information between representations and inputs. In the CPIC objective, the trade-off weight $\beta > 0$ dictates the balance between the compression and predictive information terms:

$$\min_{\psi} \mathcal{L}, \text{ where } \mathcal{L} \equiv \beta(I(X_{\text{past}}; Y_{\text{past}}) + I(X_{\text{future}}; Y_{\text{future}})) - I(Y_{\text{past}}; Y_{\text{future}}) \tag{2}$$

where ψ refer to the model parameters which encode inputs X to latent variables Y. Larger β promotes a more compact mapping and thus benefits model generalization, while smaller β leads to more predictive information in the latent space on training data. This objective function is visualized in Figure 1, where inputs X are encoded into latent space as Y via tractable encoders and the dynamics of Y are learned in a model-free manner.

The encoder $p(Y|X)$ could be implemented by fitting deep neural networks Alemi et al. (2016) to encode data X. Instead, CPIC takes an approach similar to VAEs Kingma & Welling (2013), in that it encodes data into stochastic representations. In particular, CPIC employs a stochastic encoder (g_{enc} in Figure 1) to compress input x_t into y_t as

$$y_t | x_t \sim \mathcal{N}(\mu_t, \text{diag}(\sigma_t^2)), \tag{3}$$

for each time stamp t. The mean of y_t is given by $\mu_t = g_\mu^{\text{Encoder}}(x_t)$, $U \in \mathcal{R}^{N \times D}$ whereas the variance arises from $\sigma_t = g_\sigma^{\text{Encoder}}(x_t)$.

Encoders g_μ^{Encoder} and $g_\sigma^{\text{Encoder}}$ can be any nonlinear mapping and is usually modeled using neural network architectures. We use a two-layer perceptron for a nonlinear mapping. In terms of a linear CPIC, we specify the mean of representation as $\mu_t = u^T x_t$. In both linear and nonlinear CPIC setting, if $\sigma_t = 0$, the stochastic encoder reduces to a deterministic encoder.

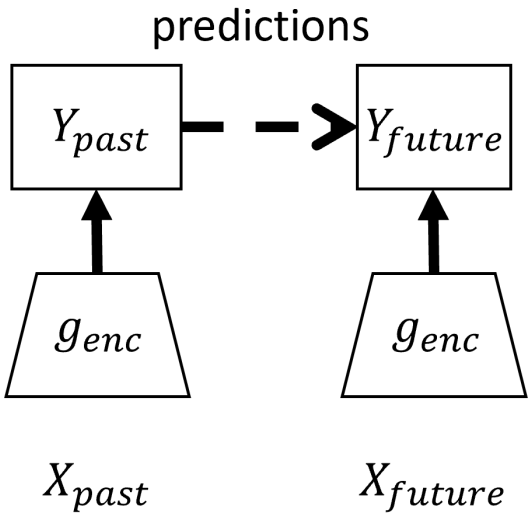

Figure 1: The overall framework of compressed predictive information coding. The encoder compress information of the input X into Y such that the predictive information between Y_{past} and Y_{future} is maximized while minimizing the mutual information between X and Y.

We extend single input to multiple inputs in the CPIC framework in terms of a specified window size T. Due to the stationary assumption, the relation between past/future blocks of input data $X(-T), X(T) \in \mathcal{R}^{N \times T}$ and encoded data $Y(-T), Y(T) \in \mathcal{R}^{Q \times T}$ are equivalent, $p(X(-T), Y(-T)) = p(X(T), Y(T))$. Without loss of generality, the compression relation can be expressed as $Y(T) = g_\mu^{\text{Encoder}}(X(T)) + \xi(T)$, where $\xi(T) \in \mathcal{N}(0, \text{blockdiag}(\text{diag}(\sigma_1^2), \ldots, \text{diag}(\sigma_T^2))$ and noise standard deviation $\sigma_t = g_\sigma^{\text{Encoder}}(x_t)$.

4 VARIATIONAL BOUNDS OF COMPRESSED PREDICTIVE INFORMATION CODING

In CPIC, since data X are stationary, the mutual information between the input data and the compressed data for the past is equivalent to that for the future $I(X(-T);Y(-T)) = I(X(T);Y(T))$. Therefore, the objective of CPIC can be rewritten as

$$\min \mathcal{L} = \beta I(X(T);Y(T)) - I(Y(-T);Y(T)) \,. \tag{4}$$

We developed the variational upper bounds on mutual information for the compression complexity $I(X(T);Y(T))$ and lower bounds on mutual information for the predictive information $I(Y(-T);Y(T))$.

4.1 UPPER BOUNDS OF COMPRESSION COMPLEXITY

In the section, we derived a tractable variational upper bound (VUB) depending on a single sample and a leave-one-out upper bound (L1Out) Poole et al. (2019) depending on multiple samples.

Theorem 1 *By introducing a variational approximation $r(y(T))$ to the marginal distribution $p(y(T))$, a tractable variational upper bound of mutual information $I(X(T);Y(T))$ is derived as $I_{VUB}(X(T);Y(T)) = \mathbb{E}_{X(T)}\big[KL(p(y(T)|x(T)), r(y(T)))\big]$.*

Theorem 2 *By utilizing a Monte Carlo approximation for variational distribution $r(y(T))$, the L1Out upper bound of mutual information $I(X(T);Y(T))$ is derived as $I_{L1Out}(X(T);Y(T)) = \mathbb{E}\left[\frac{1}{S}\sum_{i=1}^{S}\left[\log \frac{p(y(T)_i|x(T)_i)}{\frac{1}{S-1}\sum_{j\neq i} p(y(T)_i|x(T)_j)}\right]\right]$, where S is the sample size.*

The derivation details are in Appendix A and B. In practice, the L1Out bound depends on the sample size S and may suffer from numerical instability. Thus, we would like to choose the sample size S as large as possible. In general scenarios where $p(y(T)|x(T))$ is intractable, Cheng et al. (2020) proposed a variational version of VUB and L1Out by using a neural network to approximate the condition distribution $p(y(T)|x(T))$. Since the conditional distribution $p(y(T)|x(T))$ is parameterized as a known stochastic/deterministic encoder in CPIC, those variational versions are not taken into consideration.

4.2 LOWER BOUNDS OF PREDICTIVE INFORMATION

For the predictive information (PI), we derived lower bounds of $I(Y(-T);Y(T))$ using results in Agakov (2004); Alemi et al. (2016); Poole et al. (2019). In particular, we derived tractable unnormalized Barber and Agakov (TUBA) Barber & Agakov (2003) lower bounds depending on a single sample and an infoNCE lower bound Oord et al. (2018) depending on multi samples. All derivation details are discussed in Appendix C, D and E.

Theorem 3 *We derived a lower bound on predictive information (PI) I(Y(-T); Y(T)) as $I_{VLB}(Y(-T);Y(T)) = H(Y(T)) + \mathbb{E}_{p(y(-T),y(T))}[\log q(y(T)|y(-T))]$, where $q(y(T)|y(-T))$ is a variational conditional distribution.*

However, this lower bound requires a tractable decoder for the conditional distribution $q(y|x)$ Alemi et al. (2016). Alternatively we derived a TUBA Barber & Agakov (2003) lower bound which is free of the parametrization of decoder.

Theorem 4 *By introducing a differentiable critic function $f(x,y)$ and a baseline function $a(y(T))$, the TUBA lower bound of predictive information is derived as $I_{TUBA}(Y(-T),Y(T)) = \mathbb{E}_{p(y(-T),y(T))}[\tilde{f}(y(-T),y(T))] - \log\left(\mathbb{E}_{p(y(-T))p(y(T))}[e^{\tilde{f}(y(-T),y(T))}]\right)$ where $\tilde{f}(y(-T),y(T)) = f(y(-T),y(T)) - \log(a(y(T)))$.*

Different forms of the baseline function lead to different neural estimators in the literature such as MINE Belghazi et al. (2018) and NWJ Nguyen et al. (2010). On the other hand, all TUBA based estimators have high variance due to the high variance of $f(x,y)$. Oord et al. (2018) proposed a low-variance MI estimator based on noise-contrastive estimation called InfoNCE.

Theorem 5 *In the CPIC setting, the InfoNCE lower bound of predictive information is derived as*

$$I_{infoNCE}(Y(-T);Y(T)) = \mathbb{E}\left[\frac{1}{S}\sum_{i=1}^{S}\log\frac{e^{f(y(-T)_i,y(T)_i)}}{\frac{1}{S}\sum_{j=1}^{S}e^{f(y(-T)_i,y(T)_j)}}\right] \tag{5}$$

The expectation is over S independent samples from the joint distribution: $p(y(-T),y(T))$ following Markov Chain rule in Figure 1 such as $p(y((-T),y(T)) = \int p(x(-T),x(T))p(y(-T)|x(-T))p(y(T)|x(T))dx(-T)x(T)$.

4.3 Variational Bounds of CPIC

We propose two classes of lower bounds of CPIC based on whether the bounds depend on a single sample or multiple samples. We name the first class as Uni-sample lower bounds, which take the VUB upper bound of mutual information for the complexity of data compression $I(X(T),Y(T))$ and the TUBA as the lower bound of predictive information in equation 14. Thus we have

$$\mathcal{L}_{\text{UNI}} = \beta\text{KL}(p(y(T)|x(T)),r(y(T))) - I_{\text{TUBA}}(Y(-T),Y(T))\,. \tag{6}$$

Notice that by choosing different baseline functions, the lower bounds are equivalent to different mutual information estimators such as MINE and NWJ. The second class is named as multiple-sample lower bounds, which take advantage of the noise-contrastive estimation approach. The multi-sample lower bounds are expressed as

$$\mathcal{L}_{\text{MUL}} = \beta I_{\text{L1Out}}(X(T);Y(T)) - I_{\text{infoNCE}}(Y(-T);Y(T))\,. \tag{7}$$

Two main differences exist between these classes of lower bounds. First, the performance of multiple-sample lower bounds depend on batch size while uni-sample lower bounds do not. Secondly, multiple-sample lower bounds have lower variance than uni-sample lower bounds. We evaluated the performance of those variational bounds of CPIC in terms of the reconstruction performance in synthetic experiments in Appendix F, and find that with sufficiently large batch size, the multiple sample lower bound would outperform most of the uni-sample lower bounds.

Thus, without further specification, we choose the multiple-sample lower bound as the variational bound of CPIC objective in this work. Furthermore, we classify the lower bounds into stochastic and deterministic versions by whether we employ a deterministic or stochastic encoder. Notice that when choosing the deterministic encoder, the compression complexity term (first term) in equation 6 and equation 7 are constant.

5 Numerical demonstration of the superiority of CPIC

In this section, we demonstrate the superior performance of CPIC in both synthetic and real data experiments. We first examine the reconstruction performance of CPIC in noisy observations of a dynamical system (the Lorenz Attractor). The results show CPIC better recovers the latent trajectories from noisy high dimensional observations. Moreover, we demonstrate that maximizing the predictive information(PI) in the compressed latent space is more effective than maximizing PI between latent and observation space as in Creutzig & Sprekeler (2008); Creutzig et al. (2009), and also demonstrate the benefits of the stochastic representation over the deterministic representation. Secondly, we demonstrate better predictive performance of the representation evaluated by linear forecasting. The motivation for using linear forecasting models is that good representations contribute to disentangling complex data in a linearly accessible way Clark et al. (2019). Specifically, we extract latent representations and then conduct forecasting tasks given the inferred representations on two neuroscience datasets and two other real datasets. The two neuroscience datasets are multi-neuronal recordings from the hippocampus (HC) while rats navigate a maze Glaser et al. (2020) and multi-neuronal recordings from primary motor cortex (M1) during a reaching task for monkeys O'Doherty et al. (2017). The two other real datasets are multi-city temperature data (TEMP) from 30 cities over several years Gene (2017) and 12 variables from an accelerater, gyroscope, and gravity motion sensor (MS) recording human kinematics Malekzadeh et al. (2018). The forecasting tasks for the neuroscience data sets is to predict the future of the relevant exogenous variables from the past neural data, while the forecasting task for the other datasets is to predict the future of those time-series from their past. The results illustrate that CPIC has better predictive performance on these forecasting tasks compared with existing methods.

5.1 CPIC WITH STOCHASTIC ENCODING AND PI IN LATENT SPACE ROBUSTLY RECOVERS THE NOISY LORENZ ATTRACTOR

The Lorenz attractor is a 3D time series that are realizations of the Lorenz dynamical system Pchelintsev (2014). It describes a three dimensional flow generated as:

$$\frac{dx}{dt} = \sigma(y - x), \frac{dy}{dt} = f_1(\rho - z) - y, \frac{dz}{dt} = xy - \gamma z\,. \tag{8}$$

Lorenz sets the values $\sigma = 10, \beta = 8/3$ and $\rho = 28$ to exhibit chaotic behavior, as done in recent works She & Wu (2020); Clark et al. (2019); Zhao & Park (2017); Linderman et al. (2017). We simulated the trajectories from the Lorenz dynamical system and show them in the left-top panel in Figure 2. We then mapped the 3D latent signals to 30D lifted observations with a random linear embedding in the left-middle panel and add spatially anisotropic Gaussian noise on the 30D lifted observations in the left-bottom panel. The noises are generated according to different signal-to-noise ratios (SNRs), where SNR is defined by the ratio of the variance of the first principle components of dynamics and noise as in Clark et al. (2019). Specifically, we utilized 10 different SNR levels spaced evenly on a log (base 10) scale between [-3, -1] and corrupt the 30D lifted observations with noise corresponding to different SNR levels. Finally, we deploy different variants of CPICs to recover the true 3D dynamics from different corrupted 30D lifted observations with different SNR levels, and compare the accuracy of recovering the underlying Lorenz attractor time-series.

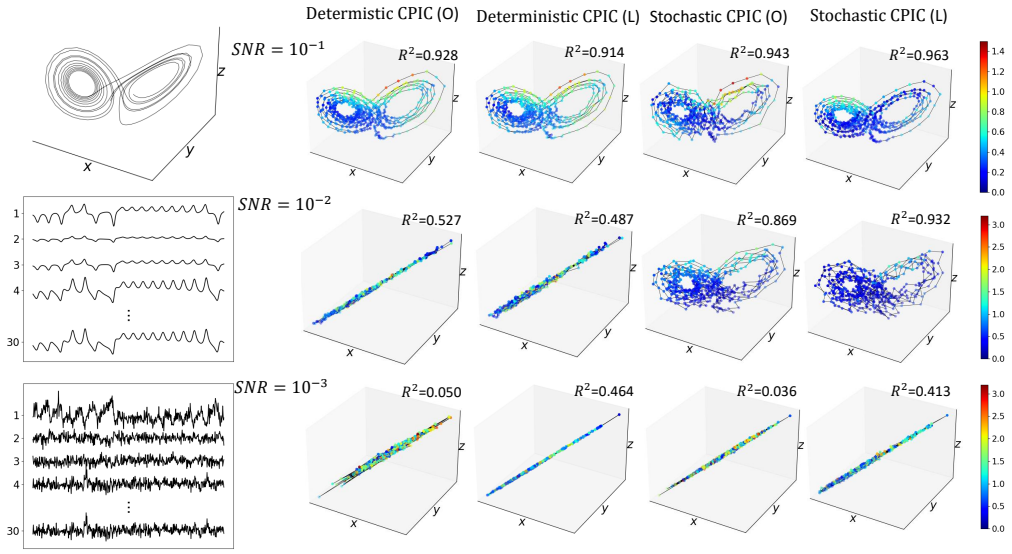

Figure 2: **Left panel.** Top: 3D trajectories of lorenz attractor's ground-truth. Middle: 30D projected trajectory. Bottom: Corrupted 30D trajectory with SNR=0.001. **Right Panel.** 3D trajectories obtained by deterministic CPIC and stochastic CPIC with PI in latent space or between latent and observation space in terms of different SNRs (0.1, 0.008, 0.001). We refer (L) to the case with PI in latent space and (O) to the case with PI between latent and observation space. We encode the point-wise Euclidean distance between the aligned inferred latent dynamics and the true dynamics into color on trajectories. Color from blue to red corresponds to the distance from short to long respectively. Separate colorbars are used for their corresponding SNRs.

We aligned the inferred latent trajectory with the true 3D dynamics with optimal linear mapping due to the reparameterization-invariant measure of latent trajectories. We validated the reconstruction performance based on the R^2 regression score of the extracted vs. true trajectories. We first compare the reconstruction performance on different variational bounds of CPIC with the latent dimension size $Q = 3$ and the time window size $T = 4$, and find that the multi-sample lower bound outperforms the uni-sample lower bounds for almost all of the 10 SNR levels. Thus, we recommend the multi-sample lower bound for CPIC in practice and use that for further results. We also find that, compared to DCA Clark et al. (2019) and CPC Oord et al. (2018) CPIC is more robust to noise and thus better

extracts the true latent trajectory from the noisy high dimensional observations. The detailed results are reported in Appendix F

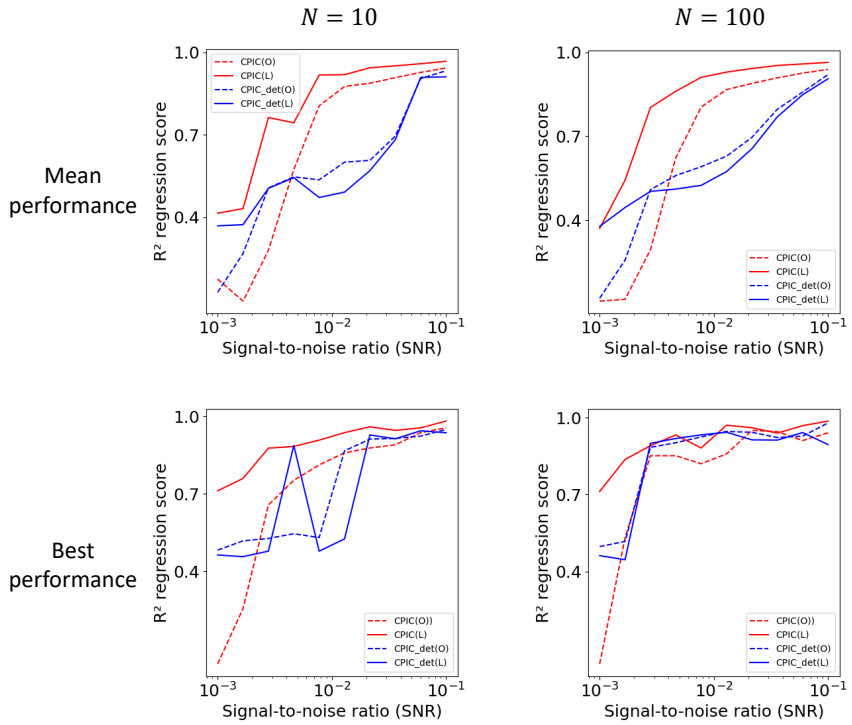

Figure 3: Comparison on R^2 scores of latent dynamics regression on 10 SNR levels for four variants of CPIC. The first row shows the mean performance of R^2 scores over running N=10/100 different random initializations and the second row shows the best performance over running N=10/100 different random initializations.

In order to demonstrate the benefits of introducing stochasticity in the encoder and maximizing the predictive information in latent space, we considered four variants of CPICs: with stochastic or deterministic encoder, and with predictive information in latent space or between latent and observation space. All four variants of CPIC models utilize the latent dimension size $Q = 3$ and the time window size $T = 4$. For each model and each SNR level, we run 100 replicates with random initializations. We show the aligned latent trajectories inferred from corrupted lifted observation for high, intermediate and low SNR (0.001, 0.01, 0.1) levels of noise with the median R^2 scores across 100 replicates in Figure 2. The point-wise distances between the recovered dynamics and the ground-truth dynamics are encoded in the colors from blue to red, corresponding to short to long distance. For high SNR (SNR = 0.1, top-right), all models did a good job of recovering the Lorenz dynamics though the stochastic CPIC with predictive information on latent space had larger R^2 than others. For intermediate SNR (SNR = 0.008, middle-right), we see that stochastic CPICs performs much better than the deterministic CPICs. Finally, as the SNR gets lower (SNR = 0.001, bottom-right) all methods perform poorly, but we note that, numerically, considering predictive information in latent space is much better than that between latent and observation space.

To more thoroughly characterize the benefits of stochastic encoding and PI in the latent space, we examined the mean of R^2 scores for the four variants on each level of SNR across $N = 10$ and $N = 100$ replicates in the top row of Figure 3. It shows that the CPIC with stochastic representations and PI in latent space robustly outperforms other variants on average. We also report the best R^2 scores for the four variants in the sense that we report the R^2 score for the model with the smallest training loss across N runs. The bottom row of Figure 3 shows that CPIC with stochastic representation and PI in latent space achieves better reconstruction and robustness to noise than other variants, especially when the number of runs N is small. Even when N is large, stochastic CPIC with PI in latent space greatly outperforms others when the noise level is high. We note that in the case of high-dimensional noisy observations with large numbers of samples common in many modern

real-world time series datasets, CPICs robustness to noise and capacity to achieve good results in a small number of runs is a clear advantage.

5.2 CPIC OUTPERFORMS OTHER METHODS IN DIVERSE REAL-WORLD FORECASTING TASKS

In this section, we show that latent representations extracted by stochastic CPIC perform better in the downstream forecasting tasks on four real datasets. We compared stochastic CPIC with contrastive predictive coding (CPC) Oord et al. (2018), PCA, SFA Wiskott & Sejnowski (2002), DCA Clark et al. (2019) and deterministic CPIC. As for CPC, we use a linear encoder for fair comparison. In addition, we compared the result from CPCs and CPICs with nonlinear encoder in which the linear mean encoder is replaced by a multi-layer perceptron. For each model, we extract the latent representations (conditional mean) and conduct prediction tasks on the relevant exogenous variable at a future time step for the neural datasets. For example, for the M1 dataset, we extract a consecutive 3-length window representation of multi-neuronal spiking activity to predict the monkey's arm position in a future time step which is lag time stamps away. The details of experiments are available in Appendix G. The predictions are conducted by linear regression [1] to emphasize the structure learned by the unsupervised methods. For these tasks, R^2 regression score is used as the evaluation metric to measure the forecasting performance. Four datasets are split into 4:1 train and test data and the forecasting task considered three different lag values (5, 10, and 15). For DCA and deterministic/stochastic CPICs, we took three different window sizes $T = 1, 2, 3$ and report the best R^2 scores. Table 1 reports all R^2 scores and demonstrates that our stochastic CPIC outperforms all other models except for the case for Temp data with forecasting at lag 15.

Table 1: Comparison between CPC-L (linear encoder), CPC-NL (non-linear encoder), PCA, SFA, DCA, D-CPIC-L (deterministic CPIC with linear encoder), S-CPIC-L (stochastic CPIC with linear encoder), D-CPIC-NL (deterministic CPIC with non-linear encoder), and S-CPIC-NL (stochastic CPIC with non-linear encoder) on R^2 regression scores on M1, Hippocampus, Temperature, and Motion sensor datasets with the optimal window size among $T \in [1, 2, 3]$ for three different lag values (5, 10, and 15). R^2 regression scores are averaged across five folds.

Dataset	Lag	CPC-L	CPC-NL	PCA	SFA	DCA	D-CPIC-L	S-CPIC-L	D-CPIC-NL	S-CPIC-NL
M1	5	0.041	0.168	0.135	0.203	0.215	0.222	0.223	0.232	**0.264**
	10	0.066	0.180	0.157	0.223	0.226	0.234	0.235	0.249	**0.291**
	15	0.068	0.152	0.145	0.199	0.200	0.202	0.203	0.226	**0.252**
HC	5	0.025	0.018	0.007	0.112	0.113	0.120	0.127	0.145	**0.150**
	10	0.012	0.012	0.001	0.101	0.101	0.107	0.113	0.121	**0.133**
	15	-0.002	0.002	-0.005	0.085	0.085	0.091	0.095	0.094	**0.114**
Temp	5	0.666	0.639	0.651	0.669	0.668	0.672	0.673	0.673	**0.673**
	10	0.630	0.584	0.615	0.630	0.632	0.629	0.633	0.630	**0.634**
	15	**0.624**	0.529	0.581	0.623	0.622	0.620	0.621	0.621	0.621
MS	5	0.281	0.184	0.107	-0.051	0.443	0.247	0.457	0.290	**0.483**
	10	0.212	0.154	0.068	-0.107	0.377	0.177	0.385	0.243	**0.425**
	15	0.182	0.136	0.044	-0.131	0.342	0.161	0.358	0.216	**0.379**

6 CONCLUDING REMARKS

We developed a novel information-theoretic framework, Compressed Predictive Information Coding, to extract representations in sequential data. CPIC balances the maximization of the predictive information in latent space with the minimization of the compression complexity of the latent representation. We leveraged stochastic representations by employing a stochastic encoder and developed variational bounds of the CPIC objective function. We demonstrated that CPIC extracts more accurate low-dimensional latent dynamics and more useful representations that have better forecasting performance in diverse downstream tasks in four real-world datasets. Together, these results indicate that CPIC will yield similar improvements in other real-world scenarios. Moreover, we note that in most real datasets, using nonlinear CPIC would lead to better representation in terms of prediction performance than linear CPIC.

[1] https://scikit-learn.org/stable/modules/linear_model.html

REFERENCES

David Barber Felix Agakov. The im algorithm: a variational approach to information maximization. *Advances in neural information processing systems*, 16(320):201, 2004.

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

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *arXiv preprint arXiv:2006.11477*, 2020.

Junwen Bai, Weiran Wang, Yingbo Zhou, and Caiming Xiong. Representation learning for sequence data with deep autoencoding predictive components. *arXiv preprint arXiv:2010.03135*, 2020.

David Barber and Felix Agakov. Information maximization in noisy channels: A variational approach. *Advances in Neural Information Processing Systems*, 16, 2003.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

William Bialek, Ilya Nemenman, and Naftali Tishby. Predictability, complexity, and learning. *Neural computation*, 13(11):2409–2463, 2001.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Gal Chechik, Amir Globerson, Naftali Tishby, Yair Weiss, and Peter Dayan. Information bottleneck for gaussian variables. *Journal of machine learning research*, 6(1), 2005.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Pengyu Cheng, Weituo Hao, Shuyang Dai, Jiachang Liu, Zhe Gan, and Lawrence Carin. Club: A contrastive log-ratio upper bound of mutual information. In *International Conference on Machine Learning*, pp. 1779–1788. PMLR, 2020.

David G Clark, Jesse A Livezey, and Kristofer E Bouchard. Unsupervised discovery of temporal structure in noisy data with dynamical components analysis. *arXiv preprint arXiv:1905.09944*, 2019.

Felix Creutzig and Henning Sprekeler. Predictive coding and the slowness principle: An information-theoretic approach. *Neural Computation*, 20(4):1026–1041, 2008.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alexander G Dimitrov, Aurel A Lazar, and Jonathan D Victor. Information theory in neuroscience. *Journal of computational neuroscience*, 30(1):1–5, 2011.

Monroe D Donsker and SR Srinivasa Varadhan. Asymptotic evaluation of certain markov process expectations for large time, i. *Communications on Pure and Applied Mathematics*, 28(1):1–47, 1975.

Selfish Gene. Historical hourly weather data. https://www.kaggle.com/datasets/selfishgene/historical-hourly-weather-data, 2017.

Joshua I Glaser, Ari S Benjamin, Raeed H Chowdhury, Matthew G Perich, Lee E Miller, and Konrad P Kording. Machine learning for neural decoding. *Eneuro*, 7(4), 2020.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

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

Alexander Lachmann, Federico M Giorgi, Gonzalo Lopez, and Andrea Califano. Aracne-ap: gene network reverse engineering through adaptive partitioning inference of mutual information. *Bioinformatics*, 32(14):2233–2235, 2016.

Scott Linderman, Matthew Johnson, Andrew Miller, Ryan Adams, David Blei, and Liam Paninski. Bayesian learning and inference in recurrent switching linear dynamical systems. In *Artificial Intelligence and Statistics*, pp. 914–922. PMLR, 2017.

Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

Mohammad Malekzadeh, Richard G Clegg, Andrea Cavallaro, and Hamed Haddadi. Protecting sensory data against sensitive inferences. In *Proceedings of the 1st Workshop on Privacy by Design in Distributed Systems*, pp. 1–6, 2018.

XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 271–279, 2016.

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

Joseph E O'Doherty, MMB Cardoso, JG Makin, and PN Sabes. Nonhuman primate reaching with multichannel sensorimotor cortex electrophysiology. *Zenodo http://doi. org/10.5281/zenodo*, 583331, 2017.

AN Pchelintsev. Numerical and physical modeling of the dynamics of the lorenz system. *Numerical analysis and Applications*, 7(2):159–167, 2014.

Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pp. 5171–5180. PMLR, 2019.

Qi She and Anqi Wu. Neural dynamics discovery via gaussian process recurrent neural networks. In *Uncertainty in Artificial Intelligence*, pp. 454–464. PMLR, 2020.

Bernard W Silverman. *Density estimation for statistics and data analysis*. Routledge, 2018.

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

Richard Turner and Maneesh Sahani. A maximum-likelihood interpretation for slow feature analysis. *Neural computation*, 19(4):1022–1038, 2007.

Weiran Wang, Qingming Tang, and Karen Livescu. Unsupervised pre-training of bidirectional speech encoders via masked reconstruction. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6889–6893. IEEE, 2020.

Yihang Wang, João Marcelo Lamim Ribeiro, and Pratyush Tiwary. Past–future information bottleneck for sampling molecular reaction coordinate simultaneously with thermodynamics and kinetics. *Nature communications*, 10(1):1–8, 2019.

Laurenz Wiskott and Terrence J Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14(4):715–770, 2002.

Yuan Zhao and Il Memming Park. Variational latent gaussian process for recovering single-trial dynamics from population spike trains. *Neural computation*, 29(5):1293–1316, 2017.

APPENDIX

A DERIVATION OF I_{VUB}

Directly estimating the compression complexity is intractable, because $I(X(T);Y(T)) := \mathbb{E}_{X(T)}\big[\mathrm{KL}(p(y(T)|x(T)), p(y(T)))\big]$ in which the population distribution $p(y(T))$ is unknown. Thus we introduce a variational approximation to the marginal distribution of encoded inputs $p(y(T))$, denoted as $r(y(T))$. Due to the non-negativity of the Kullback-Leibler (KL) divergence, the variational upper bound (VUB) is derived as

$$\begin{aligned} I(X(T);Y(T)) &= \mathbb{E}_{X(T)}\big[\mathrm{KL}(p(y(T)|x(T)), r(y(T)))\big] - \mathrm{KL}(p(y(T)), r(y(T))) \\ &\leq \mathbb{E}_{X(T)}\big[\mathrm{KL}(p(y(T)|x(T)), r(y(T)))\big] = I_{\mathrm{VUB}}(X(T);Y(T))\,. \end{aligned} \tag{9}$$

B DERIVATION OF I_{L1Out}

Generally, learning $r(y(T))$ was recognised as the distribution density estimation problem Silverman (2018), which is challenging. In this setting, the variational distribution $r(y(T))$ is assumed to be learnable, and thus estimating the variational upper bound is tractable. In particular, Alemi et al. (2016) fix $r(y(T))$ as a standard normal distribution, leading to high-bias in MI estimation. Recently, Poole et al. (2019) utilized a Monte Carlo approximation for variational distribution. In our case, with S sample pairs $(x(T)_i, y(T)_i)_{i=1}^{S}$, $r_i(y(T)) = \frac{1}{S-1}\sum_{j\neq i} p(y(T)|x(T)_j) \approx p(y(T))$ and the L1Out is derived as below:

$$I_{\mathrm{L1Out}}(X(T);Y(T)) = \mathbb{E}\left[\frac{1}{S}\sum_{i=1}^{S}\left[\log\frac{p(y(T)_i|x(T)_i)}{\frac{1}{S-1}\sum_{j\neq i} p(y(T)_i|x(T)_j)}\right]\right]\,. \tag{10}$$

C DERIVATION OF I_{VLB}

Similar to Agakov (2004), we replace the intractable conditional distribution $p(y(T)|y(-T))$ with a tractable optimization problem over a variational conditional distribution $q(y(T)|y(-T))$. It yields a lower bound on PI due to the non-negativity of the KL divergence:

$$I(Y(-T);Y(T)) \geq H(Y(T)) + \mathbb{E}_{p(y(-T),y(T))}[\log q(y(T)|y(-T))] \tag{11}$$

where $H(Y)$ is the differential entropy of variable Y and this bound is tight if and only if $q(y(T)|y(-T)) = p(y(T)|y(-T))$, suggesting that the second term in equation 11 equals the negative conditional entropy $-H(Y(T)|Y(-T))$.

However the variational lower bound requires a tractable decoder for the conditional $q(y|x)$. Alternatively, by considering an energy-based variational family for conditional distribution

The conditional expectation in equation 11 can be estimated using Monte Carlo sampling based on the encoded data distribution $p(y(-T), y(T))$. And encoded data are sampled by introducing the augmented data $x(-T)$ and $x(T)$ and marginalizing them out as

$$p(y((-T), y(T)) = \int p(x(-T), x(T))p(y(-T)|x(-T))p(y(T)|x(T))dx(-T)x(T) \tag{12}$$

according to the Markov chain proposed in Figure 1.

D Derivation of I_{TUBA}

According to Poole et al. (2019), by considering an energy-based variational family to express and conditional distribution $q(y(T)|y(-T))$:

$$q(y(T)|y(-T)) = \frac{p(y(T))e^{f(y(T),y(-T))}}{Z(y(-T))} \tag{13}$$

where $f(x, y)$ is a differentiable critic function, $Z(y(-T)) = \mathbb{E}_{p(y(T)}\left[e^{f(y(T),y(-T))}\right]$ is a partition function, and introducing a baseline function $a(y(T))$, we derived a tractable TUBA Barber & Agakov (2003) lower bound of the predictive information as:

$$\begin{aligned} I(Y(-T), Y(T)) &\geq \mathbb{E}_{p(y(-T),y(T))}[\tilde{f}(y(-T), y(T))] - \log\left(\mathbb{E}_{p(y(-T))p(y(T))}[e^{\tilde{f}(y(-T),y(T))}]\right) \\ &= I_{\text{TUBA}}(Y(-T), Y(T)) \end{aligned} \tag{14}$$

where $\tilde{f}(y(-T), y(T)) = f(y(-T), y(T)) - \log(a(y(T)))$ is treated as an updated critic function. Notice that different choices of baseline functions lead to different mutual information estimators. When $a(y(T)) = 1$, it leads to mutual information neural estimator (MINE) Belghazi et al. (2018); when $a(y(T)) = Z(y(T))$, it leads to the lower bound proposed in Donsker & Varadhan (1975) (DV) and when $a(y(T)) = e$, it recovers the lower bound in Nguyen et al. (2010) (NWJ) also known as f-GAN Nowozin et al. (2016) and MINE-f Belghazi et al. (2018). In general, the critic function $f(x, y)$ and the log baseline function $a(y)$ are usually parameterized by neural networks Oord et al. (2018); Belghazi et al. (2018): Oord et al. (2018) use a separable critic function $f(x, y) = h_\theta(x)^T g_\theta(y)$, while Belghazi et al. (2018) use a joint critic function $f(x, y) = f_\theta(x, y)$, and Poole et al. (2019) claim that joint critic function generally performs better than separable critic function but scale poorly with batch size.

E Derivation of $I_{infoNCE}$

The derivation of infoNCE in our CPIC setting is trivial by treating $Y(-T)$ and $Y(T)$ as the input and output in the infoNCE formula from the CPC setting Oord et al. (2018).

F Comparison between CPC, DCA, deterministic & stochastic CPIC (four different variational lower bounds) on R^2 regression score in the noisy Lorenz Attractor experiment

In this section, the R^2 regression scores for CPC, DCA, deterministic & stochastic CPICs (three uni-sample lower bounds: NWJ, MINE, TUBA, and one multi-sample lower bound) for all ten different SNRs are reported in Table 2. It shows that stochastic CPIC with multiple-sample lower bound outperforms other approaches in majority of SNRs. It also shows that that CPIC is most robust to the noisy data and thus detect best latent trajectories from noisy observation compared with CPC and DCA.

G Details of real-world experiments

The four real data are Monkey motor cortical dataset (M1), Rat hippocampal data (HC), Temperature dataset (Temp) and Accelerate dataset (MS).

Table 2: R^2 regression scores for CPC, DCA, deterministic & stochastic CIPCs including three uni-sample lower bounds (UNI): NWJ, MINE, TUBA, and one multi-sample lower bounds (MUL) for all ten different SNRs

SNR	CPC	DCA	CPIC							
			Deterministic				Stochastic			
			UNI			MUL	UNI			MUL
			NWJ	MINE	TUBA		NWJ	MINE	TUBA	
0.001	0.132	0.458	**0.554**	0.543	0.547	0.482	0.539	0.550	0.553	0.459
0.00167	0.195	0.466	0.539	0.538	0.574	0.430	0.573	0.569	0.571	**0.576**
0.00278	0.265	0.473	0.573	0.573	0.573	0.413	0.587	0.583	**0.590**	0.588
0.00464	0.344	0.478	0.579	0.562	0.584	0.438	**0.598**	0.583	0.556	0.593
0.00774	0.421	0.480	0.597	0.559	0.515	**0.912**	0.582	0.579	0.589	0.598
0.01292	0.491	0.484	0.587	0.596	0.597	0.468	0.580	0.563	0.592	**0.923**
0.02154	0.547	0.486	0.590	0.596	0.592	0.688	0.568	0.599	0.864	**0.930**
0.03594	0.592	0.491	0.587	0.912	0.594	0.923	0.937	0.632	0.907	**0.951**
0.05995	0.635	0.952	0.933	0.837	0.936	0.474	0.970	0.939	0.896	**0.970**
0.1	0.671	0.953	0.920	0.893	0.889	0.922	0.926	0.910	0.854	**0.989**

G.1 Monkey motor cortical dataset

O'Doherty et al. (2017) released multi-electrode spiking data for both M1 and S1 for two monkeys during a continuous grid-based reaching task. We used M1 data from the subject "Indy" (specifically, we used the file "indy_20160627_01.mat"). We discarded single units with fewer than 5,000 spikes, leaving 109 units. We binned the spikes into non-overlapping bins , square-root transformed the data and mean-centered the data using a sliding window 30 s in width.

G.2 Rat hippocampal data

Glaser et al. (2020) released the original data. The data consist of 93 minutes of extracellular recordings from layer CA1 of dorsal hippocampus while a rat chased rewards on a square platform. We discarded single units with fewer than 10 spikes, leaving 55 units. We binned the spikes into non-overlapping 50 ms bins, then square-root transformed the data.

G.3 Temperature dataset

The temperature dataset consists of hourly temperature data for 30 U.S. cities over a period of 7 years from OpenWeatherMap.org. We downsampled the data by a factor of 24 to obtain daily temperatures.

G.4 Accelerometer dataset

Malekzadeh et al. (2018) released accelerometer data which records roll, pitch, yaw, gravity x, y, z, rotation x, y, z and acceleration x, y, z for a total of 12 kinematic variables. The sampling rate is 50 Hz. We used the file "sub_19.csv" from "A_DeviceMotion_data.zip".

G.5 Forecasting task

The forecasting task is the same in Clark et al. (2019). We use the extracted consecutive 3-length window representation of endogenous data to forecast the future relevant exogenous variables at log n. In M1 and HC, the endogenous variables are processed spiking data, and the exogenous variables are location data. In Temp and MS, we assume endogenous variables and exogenous variables are the same, 30 U.S. cities' hourly temperature for Temp data and 12 kinematic variables for MS data.