SUPPLEMENTARY MATERIAL FOR CAUSAL DISCOVERY FROM CONDITIONALLY STATIONARY TIME-SERIES

Anonymous authors
Paper under double-blind review

A IMPLEMENTATION DETAILS

A.1 ENCODER ARCHITECTURE

Below we provide details of the architectures used for the encoder.

Extension from ACD The first design of the architecture extends directly from ACD (Löwe et al., 2020). In our experiments, we refer to this model as SDCI-Static.

$$\mathbf{h}_i^1 = f_{\phi_1}(\mathbf{x}_i^{1:T}) \tag{1}$$

$$\mathbf{h}_{ij}^1 = f_{\phi_2}(\mathbf{h}_i^1, \mathbf{h}_j^1) \tag{2}$$

$$\mathbf{h}_i^2 = f_{\phi_3}\Big(\sum_{i \neq j} \mathbf{h}_{ij}^1\Big) \tag{3}$$

$$\boldsymbol{\phi}_{ij} = f_{\phi_4}(\mathbf{h}_i^2, \mathbf{h}_j^2) \tag{4}$$

Figure 1 shows an overview of the structure of the model and equations from 1 to 4 denote the model computations. First, the model computes a latent embedding for each object considering the whole sequence. Then each embedding is forwarded through GNNs that caputre the inter-object correlations between the elements present in the sequence. Finally, we obtain a pairwise embedding for every pair of elements $\boldsymbol{\phi}_{ij}$ and compute the posterior distribution.More details of the architecture settings, such as the network activations or the amount of hidden layers and their size can be found in the original work from Löwe et al. (2020). The only difference of our SDCI-Static is the input size of f_{ϕ_1}, which needs to allow the one-hot representation of the state variable, and the output size of f_{ϕ_4}, which needs to generate a pairwise embedding for each of the K states as well.

The main advantage of using this architecture setting is the simplicity in implementation, since we only require to modify the input and output sizes of some parts of the model. However, we notice that the latent embedding generated when computing f_{ϕ_1} drops completely the temporal dimension. We argue this could cause the model to lose its expressiveness in capturing temporal correlations between data and therefore observe inaccurate results in its empirical study.

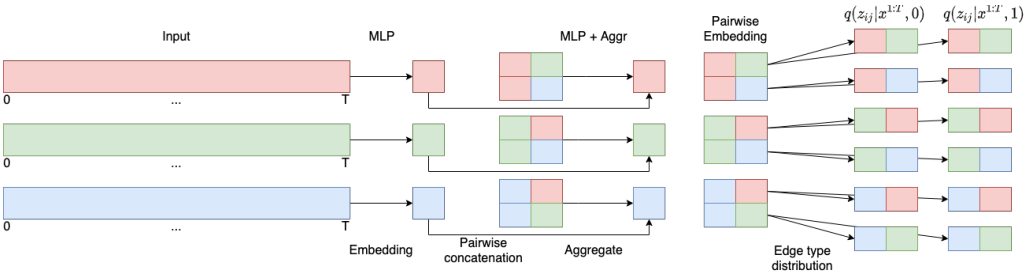

Figure 1: Illustration of the implementation of the encoder where we extend direclty from ACD (Löwe et al., 2020) and allow for conditioning on states. In the example, we consider 2 states.

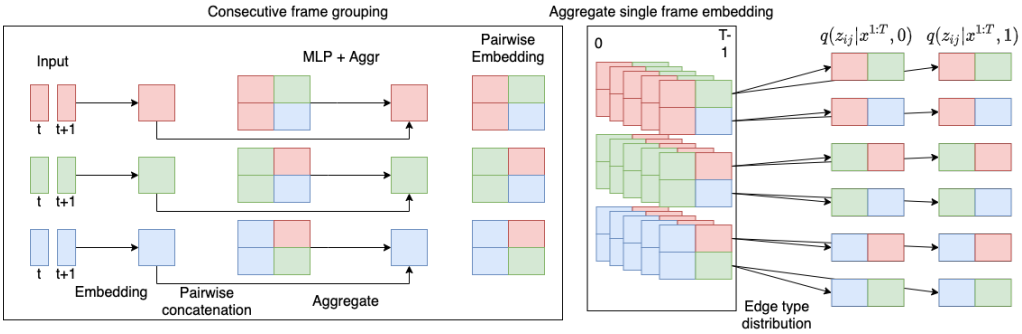

Figure 2: Illustration of the implementation of the encoder where we preserve the temporal dimension and aggregate it at last. In the example, we consider 2 states.

Preservation of temporal information The previous aspect is regarded as a potential flaw that our SDCI-Static could show when aiming to capture the *causal summary graph* of a sample. For this reason, we decide to re-design the model and preserve the temporal dimension for as long as possible. In our experiments, we refer to this model as SDCI-Temporal.

$$\mathbf{h}_i^{1,t} = f_{\phi_1}(\mathbf{x}_i^t, \mathbf{x}_i^{t+1}) \tag{5}$$

$$\mathbf{h}_{ij}^{1,t} = f_{\phi_2}(\mathbf{h}_i^{1,t}, \mathbf{h}_j^{1,t}) \tag{6}$$

$$\mathbf{h}_i^{2,t} = f_{\phi_3}\Big(\sum_{i \neq j} \mathbf{h}_{ij}^{1,t}\Big) \tag{7}$$

$$\mathbf{h}_{ij}^{2,t} = f_{\phi_4}(\mathbf{h}_i^{2,t}, \mathbf{h}_j^{2,t}) \tag{8}$$

$$\boldsymbol{\phi}_{ij} = f_{\phi_{aggr}}(\mathbf{h}_{ij}^{2,1:T-1}) \tag{9}$$

Figure 2 shows the structure of our SDCI-Temporal and equations from 5 to 9 denote the model computations. We no longer use the whole sequence at first, but concatenate consecutive frames and set it as the input to f_{ϕ_1}, we perform this computation from 1 to $T-1$ time-steps. All the subsequent steps, except for the aggregator $f_{\phi_{aggr}}$ have the same structure as the previous model.

We have considered many settings for the aggregator $f_{\phi_{aggr}}$, which aims to summarize the temporal correlations captured throughout the whole sequence. First, an MLP has been proposed. However, preliminary empirical results showed that SDCI-Temporal was not able to infer any causal structures in the data. Finally, we proposed a 1D CNN to perform the aggregation, which reported better results. The final $f_{\phi_{aggr}}$ consists of two-layer 1D CNN of 256 filters and a maxpool operation is applied in the end to erase the temporal dimension. Future work towards designing better aggregator schemes might consider attentive pooling (Lin et al., 2017), or simply perform an average pool.

B Datasets

In this section we provide detailed information about the datasets used in this work.

B.1 Linear data

Previously it has been mentioned that the generated samples produced in the linear data are unstable. However there exist many reasons why linear message passing operations have been selected as one of the datasets of this work. First, they define a simple simulated environment where one can debug and ensure that all the components are correctly implemented with ease. Furthermore, for one-dimensional variables, $p_i \in \mathbb{R}$ (which is our case), this dataset reduces to a first order Vector Autoregressive (VAR) model (Sims, 1980), which is widely used in works related to causal discovery for time-series data (Gong et al., 2015). The evolution of a sequence in this case can be expressed as follows:

$$\mathbf{p}^t = \mathbf{A}\mathbf{p}^{t-1} + \mathbf{e}^t \tag{10}$$

where $\mathbf{A}$ is the causal transition matrix and $\mathbf{e}^t$ is an independent noise process.

Regarding stability, the samples in this dataset are described by a causal transition matrix $\mathbf{A}$ where the diagonal elements are α and the off-diagonal elements are β_k where k is the edge-type interaction.For a first-order VAR to be stable, the eigenvalues of $\mathbf{A}$ need to be smaller than one in absolute value. Taking into account that each sample can obey a different underlying causal graph, one needs to check that this condition holds for all the possible arrangements of the off-diagonal elements (since the diagonal elements are always α). The number of matrices that one needs to check grows rapidly for increasing variables, which makes this computationally infeasible (recall that computing the eigenvalues of a matrix has cubic cost $O(N^3)$)

B.2 SPRING DATA

When considering springs with directed connections, we follow the generation procedure described Kipf et al. (2018) with a small modification where the spring interaction between a pair of particles can change over time (depending on the state).

In this dataset, N particles are simulated inside a 2D box where they can collide elastically with its walls. Each pair of variables is connected with uniform probability with a spring. To allow for identification of causal connections (directed edges), the connection is made unidirectional. The springs interact via the Hooke's law and this setting yield the following equations:

$$\mathbf{f}_{ij} = -\delta_k(\mathbf{r}_i - \mathbf{r}_j), \quad \ddot{\mathbf{r}}_i = \sum_{j=1}^{N} \mathbf{f}_{ij}, \quad \mathbf{p}_i = \{\mathbf{r}_i, \dot{\mathbf{r}}_i\} \tag{11}$$

where $\mathbf{f}_{ij}$ is the unidirectional interaction from particle j to particle i, δ_k denotes the edge-type for each pair of variables, and $\mathbf{r}_i$ and $\dot{\mathbf{r}}_i$ denote the 2D position and velocity of each particle. The continuous variable $\mathbf{p}_i$ is constructed by concatenating the position and the velocity.

Notice that the previous equation is defines the evolution of the continuous variable for a single time-step. In our setting, we have that $k = \mathcal{G}(s_j^t)_{ji}$. Thus, $\mathbf{f}_{ij}$ will change over time, contrary to Kipf et al. (2018). Since we consider two edge-types, we have $\delta_0 = 0$ and $\delta_1 = 0.1$. To generate samples, we need to generate a random state-dependent causal graph $\mathcal{G}(s)$ and the initial location and velocity. Then, trajectories are simulated by solving the previous differential equations using leapfrog integration. The step size used is 0.001 and the trajectories are obtained by subsampling each 100 steps.

REFERENCES

Mingming Gong, Kun Zhang, Bernhard Schoelkopf, Dacheng Tao, and Philipp Geiger. Discovering temporal causal relations from subsampled data. In *International Conference on Machine Learning*, pp. 1898–1906. PMLR, 2015.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International Conference on Machine Learning*, pp. 2688–2697. PMLR, 2018.

Zhouhan Lin, Minwei Feng, Cícero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

Sindy Löwe, David Madras, Richard S. Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. *ArXiv*, abs/2006.10833, 2020.

Christopher A Sims. Macroeconomics and reality. *Econometrica: journal of the Econometric Society*, pp. 1–48, 1980.